



- (51) International Patent Classification:
G06F 16/00 (2019.01)
- (21) International Application Number:
PCT/US2018/030390
- (22) International Filing Date:
01 May 2018 (01.05.2018)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
15/961,517 24 April 2018 (24.04.2018) US
- (71) Applicant: FACEBOOK, INC. [US/US]; 1601 Willow Road, Menlo Park, CA 94025 (US).
- (72) Inventor: SCOTT, Martin; Facebook, Inc., 1601 Willow Road, Menlo Park, CA 94025 (US).

- (74) Agent: HULSE, Robert, A. et al.; Fenwick & West LLP, 801 California Street, Mountain View, CA 94041 (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, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK,

(54) Title: USING SALIENCE RANKINGS OF ENTITIES AND TASKS TO AID COMPUTER INTERPRETATION OF NATURAL LANGUAGE INPUT

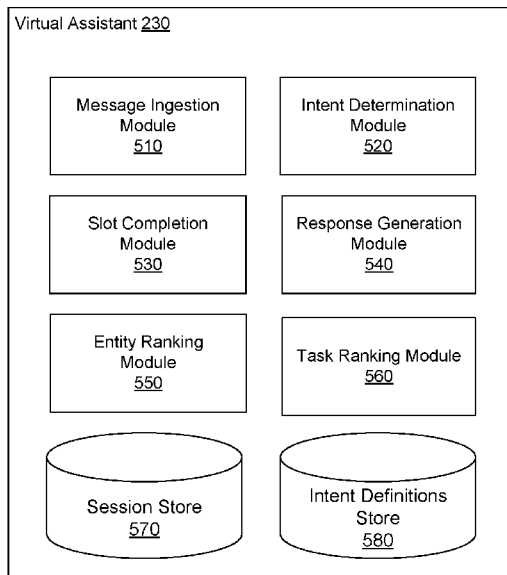


FIG. 5

(57) Abstract: A computing device receives a message that is part of a session. The message includes natural language content derived from input by a user. The computing device analyzes the natural language content to determine an intent corresponding to a given task the user desires the computing device to perform. The analyzing is based on a ranked list of tasks that indicates the current relevance of tasks in the session. A definition for the given task that includes an entity slot is retrieved from a data store and a specific entity is assigned to the slot based on a ranked list of entities that indicates current relevance of the entities in the session. The computing device invokes an agent to perform the given task based on the specific entity assigned to the entity slot.



EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV,
MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,
TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
KM, ML, MR, NE, SN, TD, TG).

Published:

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

USING SALIENCE RANKINGS OF ENTITIES AND TASKS TO AID COMPUTER INTERPRETATION OF NATURAL LANGUAGE INPUT

BACKGROUND

[0001] This disclosure relates generally to computer user interfaces and in particular to natural language-based computer interfaces.

[0002] As the availability of computing power has gone up, while the size of computing devices and associated costs have gone down, the range of uses for computing devices has increased. One area of recent growth has been virtual assistants, which are now available on many smartphones as well as through stand-alone devices. Virtual assistants interact with users to assist them in completing various tasks. For example, rather than a user opening a web browser, typing a request for the weather in the user's hometown, selecting a search result, and scrolling through the resulting webpage to determine that day's forecast, the user might just ask the assistant "what's the weather going to be like tomorrow in San Francisco?" with the intent that the virtual assistant will provide a summary of the weather forecast in response.

[0003] However, existing computer systems often struggle to correctly interpret natural language input. Where a human might rely on context and "common sense" to interpret a statement, existing computer systems often misinterpret requests, or fail to recognize them as requests at all. This leads to wasted time and frustration, as the user has to reframe the request in a more explicit way. It may even lead to the user to giving up on the natural language interface altogether and reverting to other forms of input. Improving natural language processing may improve the speed, accuracy, and usability of computer systems. This may also enable expanded and improved functionality for computerized virtual assistants, as well as other computing systems with which humans interact.

SUMMARY

[0004] A computing device (e.g., a virtual assistant) performs tasks in response to messages received from a user. The computing device uses ranked lists of actions and entities in interpreting the messages it receives. The ranked list of actions indicates the current relevance of the ranked actions in a session. Similarly, the ranked list of entities indicates the current relevance of the ranked entities in the session.

[0005] In one embodiment, the computing device receives a message that is part of the session. The message includes natural language content derived from input by the user. The computing device analyzes the natural language content based on the ranked list of tasks to determine an intent of the user, which corresponds to a task the user wishes the computing device to perform. The computing device retrieves a definition of the task from a data store that includes a slot for an entity and assigns a specific entity to the slot based on the ranked list of entities. The computing device invokes an agent to perform the task based on the entity assigned to the slot.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] FIG. 1 is a block diagram of a system environment in which a virtual assistant operates, in accordance with an embodiment.

[0007] FIG. 2 is a block diagram of the online system shown in FIG. 1, in accordance with an embodiment.

[0008] FIG. 3 is a block diagram illustrating an ontology of intents used by a virtual assistant system, in accordance with an embodiment.

[0009] FIG. 4 is a block diagram illustrating an intent node in the ontology shown in FIG. 3, in accordance with an embodiment.

[0010] FIG. 5 is a block diagram of the virtual assistant shown in FIG. 2, in accordance with an embodiment.

[0011] FIG. 6 illustrates a computer suitable for use in the system environment of FIG. 1, in accordance with an embodiment.

[0012] FIG. 7 is a flow chart illustrating a method for interpreting natural language input using salience rankings, in accordance with an embodiment.

[0013] The figures depict various embodiments for purposes of illustration only. One skilled in the art will readily recognize from the following discussion that alternative embodiments of the structures and methods illustrated herein may be employed without departing from the principles described herein.

DETAILED DESCRIPTION

OVERVIEW

[0014] When humans communicate, they do not view individual statements in isolation. Rather, a statement is interpreted in view of the context of the conversation, as well as prior

conversations between the individuals and broader social experiences. For example, consider the following conversation:

Person X: Want to have a BBQ at my place tomorrow?

Person Y: What will the weather be like?

Person X: It's meant to be sunny and about 70 degrees.

[0015] In the conversation, Person Y did not explicitly state a time or place. Even so, Person X inferred that Person Y wished to know the weather forecast for tomorrow (the time proposed for the BBQ in the previous statement) and the town in which Person X lives (knowledge inferred from previous discussions). The extent to which human communication relies on such linguistic shortcuts presents a challenge for interpreting natural language input by computers, which traditionally interpret each input statement individually. However, human users expect to be able to converse with a computerized virtual assistant in a similar manner to a human assistant. Thus, approaches that enable a computer to automatically and correctly interpret such statements may significantly improve the quality and efficiency of interactions with virtual assistants, thus increasing the utility of such technology to the user. Such approaches may also improve the utility of other computer-based technology where natural language input is desirable.

[0016] In various embodiments, a user engages in a conversation with a virtual assistant. The conversation may be verbal, in text, or use a mixture of the two. In some instances, other forms of media may be included, such as images and video. The virtual assistant maintains a pair of salience ranking lists: one for entities and one for tasks. The entity rankings and task rankings indicate a current relevance of entities and tasks, respectively, to the current conversation. The virtual assistant interprets messages received from the user based on the salience rankings list. Thus, the virtual assistant may successfully resolve ambiguous and implied references to entities or tasks. After each turn of the conversation (e.g., when the user or the virtual assistant has sent a new message), the virtual assistant updates the salience ranking lists based on the content of the message.

SYSTEM ARCHITECTURE

[0017] FIG. 1 is a block diagram of a system environment 100 for an online system 140 (e.g., a social networking system). The system environment 100 shown by FIG. 1 comprises one or more client devices 110, a network 120, one or more third-party systems 130, and the online system 140. In alternative configurations, different and/or additional components may

be included in the system environment 100. For example, the online system 140 may alternatively be a content sharing network or another system providing content to users.

[0018] The client devices 110 are one or more computing devices capable of receiving user input as well as transmitting and/or receiving data via the network 120. The client devices 110 may execute or interact with the virtual assistant 230. In one embodiment, a client device 110 is a conventional computer system, such as a desktop or a laptop computer. Alternatively, a client device 110 may be a device having computer functionality, such as a personal digital assistant (PDA), a mobile telephone, a smartphone, a tablet, or another suitable device. A client device 110 is configured to communicate via the network 120. In one embodiment, a client device 110 executes an application allowing a user of the client device 110 to interact with the online system 140. For example, a client device 110 executes a browser application to enable interaction between the client device 110 and the online system 140 via the network 120. In another embodiment, a client device 110 interacts with the online system 140 through an application programming interface (API) running on a native operating system of the client device 110, such as IOS® or ANDROID™.

[0019] The client devices 110 are configured to communicate via the network 120, which may comprise any combination of local area and/or wide area networks, using both wired and/or wireless communication systems. In one embodiment, the network 120 uses standard communications technologies and/or protocols. For example, the network 120 includes communication links using technologies such as Ethernet, 802.11, worldwide interoperability for microwave access (WiMAX), 3G, 4G, code division multiple access (CDMA), digital subscriber line (DSL), etc. Examples of networking protocols used for communicating via the network 120 include multiprotocol label switching (MPLS), transmission control protocol/Internet protocol (TCP/IP), hypertext transport protocol (HTTP), simple mail transfer protocol (SMTP), and file transfer protocol (FTP). Data exchanged over the network 120 may be represented using any suitable format, such as hypertext markup language (HTML) or extensible markup language (XML). In some embodiments, all or some of the communication links of the network 120 may be encrypted using any suitable technique or techniques.

[0020] One or more third party systems 130 may be coupled to the network 120 for communicating with the online system 140, which is further described below in conjunction with FIG. 2. In one embodiment, a third party system 130 is an application provider

communicating information describing applications for execution by a client device 110 or communicating data to client devices for use by an application executing on the client device. In other embodiments, a third party system 130 provides content or other information for presentation via a client device 110. A third party system 130 may also communicate information to the online system 140, such as advertisements, content, or information about an application provided by the third party system 130.

[0021] FIG. 2 is a block diagram of an architecture of the online system 140, according to an embodiment. The online system 140 shown in FIG. 2 includes a user profile store 205, a content store 210, an action logger 215, an action log 220, an edge store 225, a virtual assistant 230, a knowledge store 235, and a web server 240. In other embodiments, the online system 140 may include additional, fewer, or different components for various applications. For example, some embodiments may omit the knowledge store 235. Conventional components such as network interfaces, security functions, load balancers, failover servers, management and network operations consoles, and the like are not shown so as to not obscure the details of the system architecture.

[0022] Each user of the online system 140 is associated with a user profile, which is stored in the user profile store 205. A user profile includes declarative information about the user that was explicitly shared by the user and may also include profile information inferred by the online system 140. In one embodiment, a user profile includes multiple data fields, each describing one or more attributes of the corresponding online system user. Examples of information stored in a user profile include biographic, demographic, and other types of descriptive information, such as work experience, educational history, gender, hobbies or preferences, location and the like. A user profile may also store other information provided by the user, for example, images or videos. In certain embodiments, images of users may be tagged with information identifying the online system users displayed in an image, with information identifying the images in which a user is tagged stored in the user profile of the user. A user profile in the user profile store 205 may also maintain references to actions by the corresponding user performed on content items in the content store 210 and stored in the action log 220.

[0023] While user profiles in the user profile store 205 are frequently associated with individuals, allowing individuals to interact with each other via the online system 140, user profiles may also be stored for entities such as businesses or organizations. This allows an

entity to establish a presence on the online system 140 for connecting and exchanging content with other online system users. The entity may post information about itself, about its products or provide other information to users of the online system 140 using a brand page associated with the entity's user profile. Other users of the online system 140 may connect to the brand page to receive information posted to the brand page or to receive information from the brand page. A user profile associated with the brand page may include information about the entity itself, providing users with background or informational data about the entity.

[0024] The content store 210 stores objects that each represent various types of content. Examples of content represented by an object include a page post, a status update, a photograph, a video, a link, a shared content item, a gaming application achievement, a check-in event at a local business, a brand page, or any other type of content. Online system users may create objects stored by the content store 210, such as status updates, photos tagged by users to be associated with other objects in the online system 140, events, groups or applications. In some embodiments, objects are received from third-party applications or third-party applications separate from the online system 140. In one embodiment, objects in the content store 210 represent single pieces of content, or content "items." Hence, online system users are encouraged to communicate with each other by posting text and content items of various types of media to the online system 140 through various communication channels. This increases the amount of interaction of users with each other and increases the frequency with which users interact within the online system 140.

[0025] The action logger 215 receives communications about user actions internal to and/or external to the online system 140, populating the action log 220 with information about user actions. Examples of actions include adding a connection to another user, sending a message to another user, uploading an image, reading a message from another user, viewing content associated with another user, and attending an event posted by another user. In addition, a number of actions may involve an object and one or more particular users, so these actions are associated with the particular users as well and stored in the action log 220.

[0026] The action log 220 may be used by the online system 140 to track user actions on the online system, as well as actions on third party systems 130 that communicate information to the online system. Users may interact with various objects on the online system 140, and information describing these interactions is stored in the action log 220. Examples of interactions with objects include: commenting on posts, sharing links, checking-

in to physical locations via a client device 110, accessing content items, and any other suitable interactions. Additional examples of interactions with objects on the online system 140 that may be included in the action log 220 include: commenting on a photo album, communicating with a user, establishing a connection with an object, joining an event, joining a group, creating an event, authorizing an application, using an application, expressing a preference for an object (“liking” the object), and engaging in a transaction. Additionally, the action log 220 may record a user’s interactions with advertisements on the online system 140 as well as with other applications operating on the online system. In some embodiments, data from the action log 220 is used to infer interests or preferences of a user, augmenting the interests included in the user’s user profile and allowing a more complete understanding of user preferences.

[0027] The action log 220 may also store user actions taken on a third party system 130, such as an external website, and communicated to the online system. For example, an e-commerce website may recognize a user of an online system 140 through a social plug-in enabling the e-commerce website to identify the user of the online system. Because users of the online system 140 are uniquely identifiable, e-commerce websites, such as in the preceding example, may communicate information about a user’s actions outside of the online system to the online system for association with the user. Hence, the action log 220 may record information about actions users perform on a third party system 130, including webpage viewing histories, advertisements that were engaged, purchases made, and other patterns from shopping and buying. Additionally, actions a user performs via an application associated with a third party system 130 and executing on a client device 110 may be communicated to the action logger 215 by the application for recordation and association with the user in the action log 220.

[0028] In one embodiment, the edge store 225 stores information describing connections between users and other objects on the online system 140 as edges. Some edges may be defined by users, allowing users to specify their relationships with other users. For example, users may generate edges with other users that parallel the users’ real-life relationships, such as friends, co-workers, partners, and so forth. Other edges are generated when users interact with objects in the online system 140, such as expressing interest in a page on the online system, sharing a link with other users of the online system, and commenting on posts made by other users of the online system. Edges may connect two users who are connections in a

social network, or may connect a user with an object in the system. In one embodiment, the nodes and edges form a complex social network of connections indicating how users are related or connected to each other (e.g., one user accepted a friend request from another user to become connections in the social network) and how a user is connected to an object due to the user interacting with the object in some manner (e.g., “liking” a page object, joining an event object or a group object, etc.). Objects can also be connected to each other based on the objects being related or having some interaction between them.

[0029] An edge may include various features each representing characteristics of interactions between users, interactions between users and objects, or interactions between objects. For example, features included in an edge describe a rate of interaction between two users, how recently two users have interacted with each other, a rate or an amount of information retrieved by one user about an object, or numbers and types of comments posted by a user about an object. The features may also represent information describing a particular object or user. For example, a feature may represent the level of interest that a user has in a particular topic, the rate at which the user logs into the online system 140, or information describing demographic information about the user. Each feature may be associated with a source object or user, a target object or user, and a feature value. A feature may be specified as an expression based on values describing the source object or user, the target object or user, or interactions between the source object or user and target object or user; hence, an edge may be represented as one or more feature expressions.

[0030] The edge store 225 also stores information about edges, such as affinity scores for objects, interests, and other users. Affinity scores, or “affinities,” may be computed by the online system 140 over time to approximate a user’s interest in an object or in another user in the online system based on the actions performed by the user. A user’s affinity may be computed by the online system 140 over time to approximate the user’s interest in an object, in a topic, or in another user in the online system based on actions performed by the user. Multiple interactions between a user and a specific object may be stored as a single edge in the edge store 225, in one embodiment. Alternatively, each interaction between a user and a specific object is stored as a separate edge. In some embodiments, connections between users may be stored in the user profile store 205, or the user profile store 205 may access the edge store 225 to determine connections between users.

[0031] The virtual assistant 230 interacts with users via their client devices 110 to provide assistance with tasks. A user submits a request to the virtual assistant 230 via user input at a client device 110. In one embodiment, the virtual assistant 230 appears as a contact in a messaging application on the client device 110. In another embodiment, the user interacts with the virtual assistant 230 using a virtual assistant application on the client device 110. Alternatively, the virtual assistant 230 may be an application running on the client device 110.

[0032] The virtual assistant 230 maintains entity rankings and task rankings for a current session that it applies in interpreting messages received from the user. The current session may be the current conversation (e.g., any messages sent since the last period greater than a threshold length, such as five minutes, without any messages), the current calendar day, the last twenty-four hours, the last week, or any other time period. The current session may even be essentially infinite, including all messages exchanged between the user and the virtual assistant that are still available in the system environment 100. The user may also be able to define the session, such as by sending a “start new session” message to the virtual assistant 230. In some embodiments, what is considered the current session may depend on the type of messages involved in the current conversation. For example, the session for a verbal conversation might be limited to the current conversation whereas a text-based conversation might consider the session to be all messages exchanged within the last week, month, etc.

[0033] In one embodiment, the entity rankings are an ordered list of entities that have been referred to in the current session. An entity can be any tangible thing or intangible concept that the user (or the virtual assistant 230) has referred to in a message. Examples include people, locations, dates, times, objects, events, and the like. The virtual assistant 230 ranks the entities in the list based on various signals indicating the current relevance of the entity within the session.

[0034] In one embodiment, the task rankings are an ordered list of tasks that the user has asked the virtual assistant 230 to perform in the current session. Each task in the list corresponds to some action that the user might intend virtual assistant 230 to perform when sending it a message. Like entities, the tasks in the list are ranked based on signals indicative of current relevance within the session. Various embodiments of the virtual assistant 230 and the signals used to determine the rankings of entities and tasks, as well as how these rankings

are used to more effectively service user intents, are described in greater detail, with reference to FIG. 5.

[0035] FIG. 3 illustrates one embodiment of an ontology 300 of user intents. When a user sends a message to the virtual assistant 230, the user intends for the virtual assistant to perform a particular task. The virtual assistant 230 analyzes the message to determine the intent the user had for the message. In other words, the intent of the message corresponds to a task for the virtual assistant to perform. Generally there is a one-to-one correspondence between intents (what the user intends for the virtual assistant 230 to do) and tasks (what the virtual assistant does). Thus, the ranked list of tasks can help the virtual assistant 230 determine the intent of the user. For example, it is more likely that the user intends the virtual assistant 230 to perform a highly ranked task (i.e., one that is currently very relevant within the session) than a low ranking task.

[0036] In the embodiment shown in FIG. 3, the intents are divided into domains in an ontology 300. The nodes in the ontology 300 may be represented by data objects (e.g., stored in the intent definitions store 580). In the embodiment shown, the ontology 300 has a root node 310 that serves as a reference for the entire ontology. Within the ontology 300 there are several domain nodes 320, labelled D_1, D_2, \dots, D_n . A domain node 320 corresponds to a general class of tasks that a user may intend for the virtual assistant 230 to perform. For example, a first domain node 320 might be for transport services, a second domain node might be for communications services, etc. By arranging intents into domains, efficiency of the virtual assistant 230 may be improved. The hierarchical nature of the ontology 300 means that the virtual assistant 230 may first consider intents that are in the same class as the intents corresponding to recently performed tasks.

[0037] Each domain node 320 corresponds to a set of intent nodes 330, labelled I_{11}, I_{12} , etc. Different domains nodes 320 may correspond to different numbers of intent nodes 330. Each intent node 330 is a data object that defines how to perform corresponding task. For example, a domain node 320 for organizing transport services might be linked to intent nodes 330 for booking a cab, looking up a bus schedule, checking live flight arrival times, etc.

[0038] FIG. 4 shows one embodiment of an intent node 330. The intent node 330 includes a node ID 410, one or more slots 420, and one or more agents 430. In the embodiment shown, the slots 420 include a first slot 422 through an Nth slot 424 and the agents 430 include a first agent 432 through an Nth agent 434. However, in some

embodiments, the intent node 330 may include a single slot 420 and/or agent 430. An intent node 330 may be viewed as a task definition defining how a virtual assistant 230 may perform the task corresponding to the intent. In other embodiments, an intent node may include different or additional elements.

[0039] The node ID 410 identifies the node in the ontology 300. The node ID 410 may be a globally unique identifying number, an identifier of the corresponding domain node 320 and an identifying number within the name, a text string with a name of the node, or the like.

[0040] The slots 420 are variables to which the virtual assistant 230 assigns entities based on the interactions with the user and/or other available information. In one embodiment, each slot has a type corresponding to the type of entity that will fill it. Slot types include people, locations, dates, times, objects, events, and the like. Slots may have sub-types or types that overlap in scope (e.g., there may be a general “time period” slot type that can hold times, days, weeks, months, etc. and also individual slot types for times, days, weeks, months, etc.) To give an example, the slots 420 of an intent node 330 for the task of getting a weather forecast might include a location slot, a day (or range of days) slot, and a forecast granularity slot. When filled, the location slot indicates the location for which the user desires the forecast, the day slot indicates the period for which the user desires the forecast, and the forecast granularity slot indicates how detailed the forecast should be (e.g., hourly, per day, etc.). In other words, if the task is viewed as a function for the virtual assistant 230 to execute, the slots 420 are analogous to the arguments of that function.

[0041] The agents 430 fulfil the task that the intent node 330 is intended to serve. For example, if the task of an intent node 330 is to make reservations at a restaurant, the agents 430 could be applications and/or functions that submit the relevant information (day, time, number of diners, etc.) to an online booking system of the restaurant. In one embodiment, an agent (e.g., the first agent 432) is built into the intent node directly 330. In other words, the agent is a function or program that performs the desired task. In another embodiment, the agent is a pointer or other reference to a program or function that can perform the intended task. Thus, when the agent is invoked, the relevant information is sent to the agent using the pointer or reference. This may allow for the agents 430 to be updated independently from virtual assistant 230 and/or allow for third parties to provide agents to service particular intents (e.g., via a third party server 130).

[0042] Referring again to FIG. 2, the knowledge store 235 stores information that is known or assumed to be factual, according to one embodiment. This is information that the virtual assistant 230 may use to provide additional context for or generate responses to requests received from a user. In contrast to the edge store 225, which stores information about connections users and other objects that can be used as a proxy for real world connections, the knowledge store 235 stores information established from the real world. For example, if the user asks “what sci-fi movies are playing in theaters right now?” the virtual assistant 230 might retrieve a list of movie screenings from a third party system 130 and determine which of those movies are sci-fi movies based on genre information in the knowledge store 235.

[0043] The web server 240 links the online system 140 via the network 120 to the one or more client devices 110, as well as to the one or more third party systems 130. The web server 240 serves web pages, as well as other content, such as JAVA®, FLASH®, XML, and so forth. The web server 240 may receive and route messages between the online system 140 and the client device 110, for example, instant messages, queued messages (e.g., email), text messages, short message service (SMS) messages, or messages sent using any other suitable messaging technique. A user may send a request to the web server 240 to upload information (e.g., images or videos) that are stored in the content store 210. Additionally, the web server 240 may provide application programming interface (API) functionality to send data directly to native client device operating systems, such as IOS®, ANDROID™, or BlackberryOS.

EXAMPLE VIRTUAL ASSISTANT

[0044] FIG. 5 shows one embodiment of the virtual assistant 230. In the embodiment shown, the virtual assistant 230 includes a message ingestion module 510, an intent determination module 520, a slot completion module 530, a response generation module 540, an entity ranking module 550, a task ranking module 560, a session store 570, and a task definitions store 580. In other embodiments, the virtual assistant 230 contains different and/or additional elements. In addition, the functions may be distributed among the elements in a different manner than described.

[0045] The message ingestion module 510 receives messages from the client device 110 of the user. A message includes content derived from user input. The content may be natural language content. In one embodiment, the user may create messages by entering text or with verbal utterances. If the user enters text, the message may be a copy of the text entered by

the user. If the user makes a verbal utterance, the message may include an audio recording of the utterance. The message ingestion module 510 converts the audio recording to text with a speech-to-text algorithm. Alternatively, the recording may be converted to text by the client device 110 before it is sent to the virtual assistant 230. The message ingestion module 510 may also store a copy of the message (e.g., in the session store 570). Alternatively, the message may be stored later in the process (e.g., after a response has been generated).

[0046] The message ingestion module 510 parses the message to identify references to entities. In one embodiment, the message ingestion module 510 uses syntactic processing to identify words or groups of words that correspond to entities and the types of those entities. The message ingestion module 510 may also compare words in the message to an entities dictionary that includes known entities and one or more types for the entities. For example, the words “San Francisco” can be identified as a location type entity, the city of San Francisco, by comparison with a list of cities. If the message contains ambiguous references to entities, the message ingestion module 510 flags them as ambiguous. It may still be able to assign a type to the entity. For example, the pronouns “him” and “her” refer to a person-type entity, but the identity of the person is treated as ambiguous at this stage. Similarly, the word “there” is unambiguously a reference to a location-type entity, but the identity of the location is treated as ambiguous.

[0047] If the message is the first in a new session (e.g., it was preceded by a new session request or the previous message was more than a threshold amount of time in the past, etc.), the message ingestion module 510 creates new ranked lists of entities and tasks (e.g., in the session store 570). In one embodiment, the intent list is initially empty and the entities list includes just those entities that were unambiguously referenced in the message. The entities may be initially ranked in a random order, or they may be ordered based on information independent from the current session, such as user profile data indicating connections between the user and one or more of the entities. These connections may indicate the relevance of those entities to the user. For example, if the message mentions both San Francisco and New York, and the user’s profile indicates the user lives in San Francisco, San Francisco may initially be ranked above New York. In another embodiment, the initial rankings may be seeded with other entities and/or actions based on information that is independent from the session, such as the user’s profile. For example, entities that the user

has liked or is otherwise connected with on a social network may be included in the ranked list of entities.

[0048] If the message is not the first message in the session, any entities referenced that are not already on the list of entities may be added. In one embodiment, referenced entities that are not already on the ranked list are added at the bottom (i.e., with the lowest rank). Thus, when the ranked list is re-ranked after the message is processed (as described below, with reference to the entity ranking module 550), the newly referenced entities will be on the list and will likely move up in ranking due to having been referenced in the most recent message. Alternatively, the message ingestion module 510 may call the entity ranking module 550 to rank the entities before the message is processed further. This means that unambiguous references to entities in the current message may be used to help disambiguate other references in the message.

[0049] The intent determination module 520 analyzes messages received by the virtual assistant 230 to determine the intent of the user based on the ranked list of tasks. In various embodiments, the intent determination module 520 steps through the ranked list of tasks (e.g., starting with the highest ranked task) and looks for a match based on the references to entities identified in the message. To determine whether a task matches the user's intent, the intent determination module 520 retrieves the intent node 430 corresponding to the task (e.g., from intent definitions storage 580) and compares the types of the slots 420 of the node with the types of the entities identified in the message.

[0050] In one embodiment, the intent determination module 520 determines whether all of the slots 420 can be filled by an entity of the appropriate type referenced in the message (either ambiguously or unambiguously). If so, the intent corresponding to that task is assigned to the message. If not, the intent determination module 520 considers the next most highly ranked task to determine whether it is a match, and so on, until a match is found or the end of the ranked list of tasks is reached. If the end of the ranked list of tasks is reached without finding a match, the intent determination module 520 sends a message to the user's client device 110 requesting clarification of the user's intent. For example, the message might state "I'm sorry, I didn't understand that. Please ask again."

[0051] In another embodiment, the intent determination module 520 also considers whether some or all of the slots 420 can be filled inferentially based on the ranked list of entities. If the message does not include an entity of the appropriate type to fill a slot 420,

but there is an appropriate entity in the ranked list of entities, the intent determination module 520 fills the slot with that entity. In this case, the intent determination module 520 generates a confidence value indicating how likely it is that the task being considered corresponds to the intent of the user. If the confidence value exceeds an intent confidence threshold (e.g., 85%), the intent determination module 520 may assign that intent to the message.

[0052] Alternatively, the intent determination module 520 may calculate a confidence value for some (e.g., the top five) or all of the other tasks in the ranked list. The confidence values are calculated by combining the ranking of the task with the degree to which the identified entities match the slots 420 for the intent corresponding to the task. For example, each intent under consideration may be assigned a composite ranking based on the fitness of the task for the intent based on the message and a score for the fitness of each of the identified entities for one of the slots in the intent. These entity/slot compatibility scores may be determined heuristically or via machine-learned weights. In either case, features such as the slot and entity types may be considered, along with the position of the entity in the ranked list. If no task has a confidence value that exceeds the intent confidence threshold or more than one task has a confidence value exceeding the threshold, the intent determination module 520 sends a message to the user's client device 110 asking for clarification. The message may identify the most likely task (e.g., the one with the highest confidence value or the one with the highest ranking) and ask for confirmation. For example, the message might read "Do you want me to get you the weather forecast for tomorrow in San Francisco?"

[0053] In some embodiments, the intent determination module 520 starts from the presumption that the current intent will be in the same domain as the intent corresponding to the previous task performed by the virtual assistant 230 (e.g., as indicated in the session store 570). The intent determination module 520 identifies the domain of the previous task (e.g., by querying the ontology for the node ID 410 corresponding to the previous task) and filters the ranked list of actions for those in the same domain. The intent determination module 520 considers actions corresponding to intents in that domain first (in ranked order). If no match is found in the current domain, the intent determination module 520 then considers actions from the ranked list corresponding to intents in other domains (again, in ranking order).

[0054] Assuming that a task has been selected, the slot completion module 530 fills in the slots 420 of the task definition (e.g., as retrieved from the corresponding intent node 330 in the intent definition storage 580) using the entities identified in the message and the ranked

list of entities. This may involve resolving ambiguous references in the message (e.g., to whom does a pronoun refer?) and/or determining implicit references (e.g., in saying “what’s the weather like?” the user wishes to know about the weather in the place he or she referenced in a previous message, or that was the result of a previous task).

[0055] In one embodiment, for each slot where the message includes an ambiguous reference or there is no suitable reference, the slot completion module 530 attempts to fill the slot based on the ranked list of entities. The slot completion module 530 steps through the ranked list of entities (starting with the highest ranked) and fills the slot with the first entity that has a type that matches the slot’s type. The types of the entities may be stored in the ranked list or may be determined using syntactic analysis and/or an entities dictionary. For example, if the slot is for a location, then the highest ranked location in the list is selected. In another embodiment, the slot completion module 530 filters the ranked list of entities first to remove those that are not of an appropriate type for the slot. This may improve efficiency as highly ranked entities that are not of an appropriate type for the slot need not be considered. For example, if the slot is for a person, any entities in the ranked list that are not people need not be considered.

[0056] In some embodiments, the slot completion module 530 generates confidence values indicating how likely it is that some (e.g., the top five ranked entities) or all of the entities in the ranked list match the slot. The confidence value for an entity may be calculated by combining the ranking of the entity, a score generated from syntactic analysis of the message, and/or information that is independent from the current session (e.g., user profile data). For example, in the case of a location entity, the confidence values may be derived from geographic proximity to a current location of the user that sent the message. In the case of a person entity, the confidence values may be derived from the relationship between the user and the entity in question (e.g., as indicated in a social graph edge). The confidence values may also be influenced by how recently the entity in question has been mentioned as well as how many times it has been mentioned in the current conversational thread. These values may either be determined heuristically or via weights obtained from a machine-learned model. If no entity has a confidence value that exceeds an entity confidence threshold (e.g., 85%), the slot completion module 530 sends a message to the user’s client device 110 asking for clarification. The message may identify the most likely entity (e.g., the one with the highest confidence value or the highest ranking) and ask for confirmation. For

example, the message might read “Are you referring to John Smith?” If two or more intents have confidence values that exceed the threshold, the slot completion module 530 may also request clarification from the user. For example, where the message from user referred to “John,” the slot completion module 350 might send a message that reads “Are you referring to John Smith or John Jones?”

[0057] The response generation module 540 generates responses to messages received from the client device 110 of the user. In various embodiments, the response generation module 540 uses the intent determined by the intent determination module 540 and the slot values determined by the slot completion module 540 to identify an appropriate agent and formulate a request to that agent. As described previously, an intent node 330 in an intent ontology 300 may include a set of one or more agents 430. Where there is more than one agent that may serve the intent, the response generation module 540 may select one based on various considerations, including preferences of the user and the slot values.

[0058] In one embodiment, the response generation module 540 prompts the user to select an agent from the set of agents 430 the first time the user expresses that intent. In future, the response generation module 540 may automatically select the same agent unless the user expressly overrides the previous selection (e.g., by sending a message to the virtual assistant 230 requesting that the previous agent selection be forgotten). Alternatively or additionally, the response generation module 540 may consider specific slot values in selecting an agent. One agent in an intent node 330 may be preferable for servicing the intent in relation to one entity, while another agent may be preferable in relation to a different entity. For example, a first agent may be better (or preferred by the user) for making restaurant reservations in San Francisco while a second agent may be better (or preferred by the user) when making restaurant reservations in New York.

[0059] Regardless of how the agent is selected, the response generation module 540 invokes the selected agent using the values of the slots (as determined by the slot completion module 530) as input. In one embodiment, assuming the slot values define a viable invocation request, the agent returns the requested result. For example, if the agent is for obtaining weather information, it will return a weather forecast for the requested location at the requested time, etc. As another example, if the agent is for making restaurant reservations, it will return confirmation that the reservation has been made (assuming it was successful). In contrast, if the invocation request is not viable (e.g., because the virtual

assistant 230 assigned an invalid value to one of the slots) or if the request cannot be fulfilled for some other reason (e.g., the request was for a restaurant reservation and no tables are available) then the agent may return an error.

[0060] The response generation module 540 creates a message to send to the user's client device 110 based on the result received from the agent. The response generation may use natural language generation techniques such that the content of the message is constructed in a way that is similar or identical to that which a human would create. If the agent successfully completed the intended task, the message may include details of the result (e.g., confirmation of a restaurant booking, a weather forecast, etc.). In contrast, if invocation of the agent failed, the message may seek clarification from the user as to what task the user wishes the virtual assistant 230 to perform. Alternatively, the invoked agent may return the result directly to the user's client device 110.

[0061] The entity ranking module 550 updates the ranked list of entities a session based on messages received by the virtual assistant 230 and/or responses sent to the user's client device 110 during that session. In one embodiment, the entity ranking module 550 ranks the entities in the list after each turn of a conversation (e.g., after each message is received and each response is sent). Alternatively, the ranking may be performed at other intervals, such as after each message sent by the user is processed. Regardless of the interval used, the entity ranking module 550 updates the ranked list by calculating ranking scores for the entities and ordering the entities based on the ranking scores.

[0062] In one embodiment, the entity ranking module 550 calculates the ranking score for each entity in the list based on a set of features. The set of features can include: a number of mentions of that entity during the current session; a measure of recency of the entity in the session (e.g., the number of turns since the last mention of the entity); profile information of the user indicating the importance of the entity to the user; contextual prominence of the entity (e.g., a number of mentions of the entity on a social network site in the last twenty-four hours, connections between the entity and other entities mentioned recently in the session, etc.); a frequency of interaction between the user and the entity in a messaging application (where the entity is of a type that has such an account); and a measure of how close the user is to the entity in a social network (e.g., where the entity is another user, how many friends and/or interests the user and the entity have in common on a social networking site). For example, the measure of how close the user is to a person entity with an account on the same

social network could be derived by taking the number of connections shared by both the user and the entity in question and dividing it by the total number of connections of the user.. In other embodiments, the set of features may include different or additional features.

[0063] Regardless of the specific features used, the entity ranking module 550 generates a ranking score for an entity based on the values of those features. In one embodiment, a set of heuristically predetermined weights are assigned to the features, with the score being the sum of the features multiplied by the corresponding weights. In another embodiment, a machine-learning model is used to determine the weight for each feature.

[0064] The machine-learning model may be trained using a set of training data. For example, a set of human operators might label training a set of conversations between a user and a virtual assistant with scores indicating the relative importance of various entities in the conversation. The model (e.g., a neural network) is trained (e.g., using a backpropagation algorithm) to produce scores for the training set that closely match the human-assigned scores. The model may then be validated by applying it to a second set of conversations and comparing the results to those assigned by the human-operators for the second set.

[0065] Regardless of the specific method used to generate the ranking scores, the entity ranking module 550 ranks the list of entities based on the ranking scores. The ranked list of entities provides a measure of how significant each entity currently is to the conversation, with higher ranking scores indicating that the entity is likely more significant and vice versa. A highly ranked entity (with a high ranking score) might have been mentioned several times, including in the most recent message, and the user may be connected in some way with the entity on a social network. Conversely, a lowly ranked entity (with a low ranking score) might have been mentioned only once several messages back in the conversation and have no other apparent connections to the user.

[0066] The task ranking module 560 operates similarly to the entity ranking module 550, but ranks tasks rather than entities. In one embodiment, the task ranking module 560 generates a ranking score for each task from a set of features that includes: a number of invocations of the task in the current session; a measure of recency for the task (e.g., the number of turns since the last invocation of the task); the percentage of the last n tasks invoked that were that task (where n is a positive integer, such as ten); the prominence of the task in the user's geographic area (e.g., how many times has the task been invoked in the user's ZIP code in the last week); the prominence of the task among other users associated

with the user (e.g., the user's friends within a social network, the user's contacts in an instant messaging service, users who share a messaging thread with the user and the virtual assistant 230, etc.); and a measure of the relevance of the task to the types of entities currently included in the entities list (or the top m ranked entities, where m is a positive integer, e.g., ten). For example, the relevance of the task to the types of entities currently included in the entities list may be determined by incrementing the score for the task for every entity in the ranked list whose type matches the type of a slot included in the definition of that task. The amount the score is incremented for a given entity may depend on the ranking of the entity. Higher-ranked entities can provide a larger increase in the score by applying a multiplier (e.g., the reciprocal of the entity's position in the ranked list). The contribution of each signal to the ranking score may be determined based on a predetermined heuristic weighting or using a machine-learning approach, as described previously with reference to the entity ranking module 550.

[0067] Regardless of how they are determined, the task ranking module 560 ranks the tasks based on the ranking scores. Thus, the ranked list of tasks provides a measure of how significant each task currently is to the conversation, with higher ranking scores indicating that the task is likely more significant and vice versa. For example, if the user asked about the weather tomorrow in a particular town and then asked about the price to stay at various hotels in the town that night, a "get hotel price" task might be highly ranked whereas a "get weather forecast" task might be lower ranked.

[0068] The session store 570 includes one or more computer-readable media configured to store information about the current session. In one embodiment, the information about the current session includes the ranked lists of entities and tasks as well as a copy of each message sent and received by the virtual assistant 230 in the session. The session store 570 may also store copies of messages exchanged during previous sessions. Although the session store 570 is shown as a single entity located within the virtual assistant 230, it may be located elsewhere and/or distributed across multiple computing devices. For example, the session store 570 may be a distributed database stored across multiple remote servers.

[0069] The intent definitions store 580 includes one or more computer-readable media configured to store information about the intents serviced by the virtual assistant 230. In one embodiment, the information about the intents is an intent hierarchy 300, such as the one shown in FIG. 3. Like the session store 570, the intent definitions store 580 may be remote

from the virtual assistant and/or distributed across multiple computing devices. In addition, although the session store 570 and intent definitions store 580 are shown in FIG. 5 as distinct components, the corresponding functionality may be provided by a single database or other type of data store.

EXAMPLE COMPUTING SYSTEM ARCHITECTURE

[0070] FIG. 6 is a high-level block diagram illustrating an example computer 600 suitable for use as a client device 110, third party system 130, or server in the online system 140. The example computer 600 includes at least one processor 602 coupled to a chipset 604. The chipset 604 includes a memory controller hub 620 and an input/output (I/O) controller hub 622. A memory 606 and a graphics adapter 612 are coupled to the memory controller hub 620, and a display 618 is coupled to the graphics adapter 612. A storage device 608, keyboard 610, pointing device 614, and network adapter 616 are coupled to the I/O controller hub 622. Other embodiments of the computer 600 have different architectures.

[0071] In the embodiment shown in FIG. 6, the storage device 608 is a non-transitory computer-readable storage medium such as a hard drive, compact disk read-only memory (CD-ROM), DVD, or a solid-state memory device. The memory 606 holds instructions and data used by the processor 602. The pointing device 614 is a mouse, track ball, touch-screen, or other type of pointing device, and is used in combination with the keyboard 610 (which may be an on-screen keyboard) to input data into the computer system 600. The graphics adapter 612 displays images and other information on the display 618. The network adapter 616 couples the computer system 400 to one or more computer networks.

[0072] The types of computers used by the entities of FIGS. 1, 2, and 5 can vary depending upon the embodiment and the processing power required by the entity. For example, the online system 140 might include a distributed database system comprising multiple blade servers working together to provide the functionality described. Furthermore, the computers can lack some of the components described above, such as keyboards 610, graphics adapters 612, and displays 618.

EXAMPLE METHOD OF INTERPRETING HUMAN INPUT USING SALIENCE RANKINGS

[0073] FIG. 7 illustrates one embodiment of a method 700 for interpreting natural language input using salience rankings. The steps of FIG. 7 are illustrated from the perspective of the virtual assistant 230 performing the method. However, some or all of the steps may be performed by other entities and/or components. In addition, some embodiments

may perform the steps in parallel, perform the steps in different orders, or perform different steps. For example, the virtual assistant may update the salience rankings based on a received message before or in parallel to generating a response to the message.

[0074] In the embodiment shown in FIG. 7, the method 700 begins with the virtual assistant 230 receiving 710 a message from a client device 110. The message includes content generated from user input. The content can include be text (e.g., typed on a keyboard) and/or audio data of the user talking (e.g., recorded by a microphone of the client device 110). Where the user input includes audio data, the virtual assistant 230 converts it to text (e.g., with a speech-to-text algorithm). The content may additionally include other types of media, such as images and video.

[0075] The virtual assistant 230 interprets 720 the message based on the current salience rankings. In one embodiment, the virtual assistant 230 determines the intent of the message based on a ranked list of tasks and resolves any ambiguous references to entities (pronouns, a name that matches more than entity, etc.) based on a ranked list of entities. As described previously, the ranked list of tasks indicates the relevance of each task to the current session. Similarly, the ranked list of entities indicates the relevance of different entities to the current session.

[0076] The virtual assistant 230 invokes 730 an agent to complete the task corresponding to the identified intent. In one embodiment, the virtual assistant 230 retrieves the intent's definition (e.g., from an intent definitions store 380), which defines how the assistant may perform corresponding task. The definition includes one or more slots to be filled by an entity. The virtual assistant fills the slots based on the ranked list of entities and invokes 730 the agent to complete the task. The specific agent to invoke may be included or identified in the definition. The agent completes the task and returns the result to the virtual assistant 230.

[0077] The virtual assistant 230 generates 740 a response to the message. The response includes the result returned by the agent. Alternatively, the agent may return the result directly to the user's client device 110. The virtual assistant 230 provides 750 the response for presentation to the user. In one embodiment, providing 750 the response includes sending the response to a client device 110 associated with the user (e.g., the device from which the message was received). The client device 110 then displays the result of invoking the agent to the user (e.g., by displaying it within an app with which the user communicates with the virtual assistant 230).

[0078] The virtual assistant 230 also updates 760 the salience rankings based on the received message, as interpreted 720. In one embodiment, the virtual assistant 230 adds the message to the session history (e.g., as stored in the session store 570) and re-ranks the lists of tasks and entities. In another embodiment, the virtual assistant 230 adjusts just the ranking scores of the task performed in the current turn and any entities referenced in the message. This may be more efficient as less calculation is required, but may also reduce accuracy of the rankings overall. In such an embodiment, the virtual assistant may periodically (e.g., every ten turns) recalculate the ranked lists entirely to balance efficiency with accuracy.

[0079] A user interface using the described salience rankings can provide several advantages. One such advantage is improved intent-level disambiguation. This arises where the user asks about an entity that could be one of several things (e.g., a book, a play, or a movie, all with the same title). The task rankings can assist the virtual assistant 230 in correctly identifying the correct identity without having to seek explicit clarification from the user. For example, if the user has recently asked about two other books, it is most likely that the intent of the ambiguous request was also to ask about a book, not a play or movie. Thus, the efficiency of the user interface can be improved as the virtual assistant 230 may make less mistakes in interpreting requests and/or may not need to request clarification from the user as often.

[0080] Another advantage is improved coreference resolution. This arises where the dialog has recently included references to two entities and the user then makes a request including a pronoun. The entity rankings can improve the ability of the virtual assistant 230 to correctly determine which entity the user is referring to. This may be the case even where the most recent reference was to the incorrect entity if the sum of the session history indicates that the correct entity is more important to the overall conversation. Thus, as with intent-level disambiguation, the efficiency of the user interface may be improved.

[0081] A similar advantage that may be realized is slot carryover. This refers to a situation where the user has been asking about one thing and briefly segues to another before returning to the first topic. For example, the user might ask several questions about a restaurant, then ask whether it will be raining that night, and then ask “are there any gluten-free options?” A human easily interprets the last question as referring to the restaurant not the weather forecast for the corresponding location. The entity rankings may enable the virtual assistant 230 to correctly determine the user’s intent because the overall prominence

of the restaurant in the session history may override the more recent reference to the weather. Similarly, the ranked list of tasks may enable the virtual assistant 230 to successfully interpret the user returning to a previous intent after a brief interlude into another topic. For example, if the user requests a list of available hotels in various ski resorts, then asks about the weather at one of the ski resorts, and then asks “how about Tahoe?” the virtual assistant 230 may correctly interpret this as a request for hotel availability, not a request about the weather, based on the overall prominence of the hotel availability task in the session history. Therefore, the efficiency of the user interface may be improved. The user can submit requests in an abbreviated, natural language form and the virtual assistant may require less clarifications and/or make less errors.

CONCLUSION

[0082] The foregoing description of the embodiments has been presented for the purpose of illustration; it is not intended to be exhaustive or to limit the patent rights to the precise forms disclosed. Persons skilled in the relevant art can appreciate that many modifications and variations are possible in light of the above disclosure.

[0083] Some portions of this description describe the embodiments in terms of algorithms and symbolic representations of operations on information. These algorithmic descriptions and representations are commonly used by those skilled in the data processing arts to convey the substance of their work effectively to others skilled in the art. These operations, while described functionally, computationally, or logically, are understood to be implemented by computer programs or equivalent electrical circuits, microcode, or the like. Furthermore, it has also proven convenient at times, to refer to these arrangements of operations as modules, without loss of generality. The described operations and their associated modules may be embodied in software, firmware, hardware, or any combinations thereof.

[0084] Any of the steps, operations, or processes described herein may be performed or implemented with one or more hardware or software modules, alone or in combination with other devices. In one embodiment, a software module is implemented with a computer program product comprising a computer-readable medium containing computer program code, which can be executed by a computer processor for performing any or all of the steps, operations, or processes described.

[0085] Embodiments may also relate to an apparatus for performing the operations herein. This apparatus may be specially constructed for the required purposes, and/or it may

comprise a general-purpose computing device selectively activated or reconfigured by a computer program stored in the computer. Such a computer program may be stored in a non-transitory, tangible computer readable storage medium, or any type of media suitable for storing electronic instructions, which may be coupled to a computer system bus.

Furthermore, any computing systems referred to in the specification may include a single processor or may be architectures employing multiple processor designs for increased computing capability.

[0086] Embodiments may also relate to a product that is produced by a computing process described herein. Such a product may comprise information resulting from a computing process, where the information is stored on a non-transitory, tangible computer readable storage medium and may include any embodiment of a computer program product or other data combination described herein.

[0087] Finally, the language used in the specification has been principally selected for readability and instructional purposes, and it may not have been selected to delineate or circumscribe the patent rights. It is therefore intended that the scope of the patent rights be limited not by this detailed description, but rather by any claims that issue on an application based hereon. Accordingly, the disclosure of the embodiments is intended to be illustrative, but not limiting, of the scope of the patent rights, which is set forth in the following claims.

CLAIMS

What is claimed is:

1. A method comprising:
 - receiving a message at a computing device, the message part of a session and including natural language content derived from input by a user;
 - analyzing the natural language content to determine an intent, the intent corresponding to a given task the user desires the computing device to perform, the analyzing based on a ranked list of tasks that indicates current relevance of the tasks in the session;
 - retrieving, from a data store, a definition for the given task, the definition including an entity slot;
 - assigning a specific entity to the entity slot based on a ranked list of entities that indicates current relevance of the entities in the session; and
 - invoking an agent to perform the given task, the agent performing the given task based on the specific entity assigned to the entity slot.
2. The method of claim 1, further comprising updating the ranked list of tasks based on the message, updating the ranked list of tasks including generating a ranking score for a task based on a plurality of features, the plurality of features including a number of invocations of the task in the current session and a measure of recency for the task in the current session.
3. The method of claim 1, wherein determining the intent comprises:
 - calculating confidence values for a plurality of tasks that includes the given task, each confidence value indicating a likelihood that the corresponding task corresponds to the intent; and
 - selecting the given task based on the confidence values.
4. The method of claim 1, wherein determining the intent comprises:
 - calculating confidence values for a plurality of tasks, each confidence value indicating a likelihood that the corresponding task corresponds to the intent; and
 - sending a request to a computing device of the user responsive to none of the confidence values exceeding an intent confidence threshold, the request asking the user to clarify the intent.

5. The method of claim 1, wherein the entity slot has a slot type and assigning the specific entity to the entity slot comprises:

considering entities in the ranked list of entities in rank order; and
selecting, as the specific entity, a highest-ranked entity that has an entity type that matches the slot type.

6. The method of claim 1, the method further comprising updating the ranked list of entities based on the message, updating the ranked list of entities including generating a ranking score for an entity based on a plurality of features, the plurality of features including a number of mentions of the entity during the current session and a measure of recency for the entity in the session.

7. The method of claim 6, wherein the plurality of features further includes information about the user that is independent from the current session, the information about the user indicating a relevance of the entity to the user.

8. A non-transitory computer-readable medium storing computer program instructions executable by a processor to perform operations comprising:

receiving a message at a computing device, the message part of a session and including natural language content derived from input by a user;
analyzing the natural language content to determine an intent, the intent corresponding to a given task the user desires the computing device to perform, the analyzing based on a ranked list of tasks that indicates current relevance of the tasks in the session;
retrieving, from a data store, a definition for the given task, the definition including an entity slot;
assigning a specific entity to the entity slot based on a ranked list of entities that indicates current relevance of the entities in the session; and
invoking an agent to perform the given task, the agent performing the given task based on the specific entity assigned to the entity slot.

9. The non-transitory computer-readable medium of claim 8, wherein the operations further comprise updating the ranked list of tasks based on the message, updating the ranked list of tasks including generating a ranking score for a task based on a plurality of features, the plurality of features including a number of invocations of the task in the current session and a measure of recency for the task in the current session.

10. The non-transitory computer-readable medium of claim 8, wherein determining the intent comprises:
 - calculating confidence values for a plurality of tasks that includes the given task, each confidence value indicating a likelihood that the corresponding task corresponds to the intent; and
 - selecting the given task based on the confidence values.
11. The non-transitory computer-readable medium of claim 8, wherein determining the intent comprises:
 - calculating confidence values for a plurality of tasks, each confidence value indicating a likelihood that the corresponding task corresponds to the intent; and
 - sending a request to a computing device of the user responsive to none of the confidence values exceeding an intent confidence threshold, the request asking the user to clarify the intent.
12. The non-transitory computer-readable medium of claim 8, wherein the entity slot has a slot type and assigning the specific entity to the entity slot comprises:
 - considering entities in the ranked list of entities in rank order; and
 - selecting, as the specific entity, a highest-ranked entity that has an entity type that matches the slot type.
13. The non-transitory computer-readable medium of claim 8, wherein the operations further comprise updating the ranked list of entities based on the message, updating the ranked list of entities including generating a ranking score for an entity based on a plurality of features, the plurality of features including a number of mentions of the entity during the current session and a measure of recency for the entity in the session.
14. The non-transitory computer-readable medium of claim 13, wherein the plurality of features further includes information about the user that is independent from the current session, the information about the user indicating a relevance of the entity to the user.
15. A computer system comprising:
 - one or more processors; and
 - one or more computer readable media storing computer program instructions that, when executed, cause the one or more processors to perform operations including:

receiving a message at a computing device, the message part of a session and including natural language content derived from input by a user;

analyzing the natural language content to determine an intent, the intent corresponding to a given task the user desires the computing device to perform, the analyzing based on a ranked list of tasks that indicates current relevance of the tasks in the session;

retrieving, from a data store, a definition for the given task, the definition including an entity slot;

assigning a specific entity to the entity slot based on a ranked list of entities that indicates current relevance of the entities in the session; and

invoking an agent to perform the given task, the agent performing the given task based on the specific entity assigned to the entity slot.

16. The computer system of claim 15, wherein the operations further include updating the ranked list of tasks based on the message, updating the ranked list of tasks including generating a ranking score for a task based on a plurality of features, the plurality of features including a number of invocations of the task in the current session and a measure of recency for the task in the current session.

17. The computer system of claim 15, wherein determining the intent comprises: calculating confidence values for a plurality of tasks that includes the given task, each confidence value indicating a likelihood that the corresponding task corresponds to the intent; and selecting the given task based on the confidence values.

18. The computer system of claim 15, wherein determining the intent comprises: calculating confidence values for a plurality of tasks, each confidence value indicating a likelihood that the corresponding task corresponds to the intent; and sending a request to a computing device of the user responsive to none of the confidence values exceeding an intent confidence threshold, the request asking the user to clarify the intent.

19. The computer system of claim 15, wherein the entity slot has a slot type and assigning the specific entity to the entity slot comprises:

considering entities in the ranked list of entities in rank order; and
selecting, as the specific entity, a highest-ranked entity that has an entity type that
matches the slot type.

20. The computer system of claim 15, wherein the operations further include updating the ranked list of entities based on the message, updating the ranked list of entities including generating a ranking score for an entity based on a plurality of features, the plurality of features including a number of mentions of the entity during the current session, a measure of recency for the entity in the session, and information about the user that is independent from the current session, the information about the user indicating a relevance of the entity to the user.

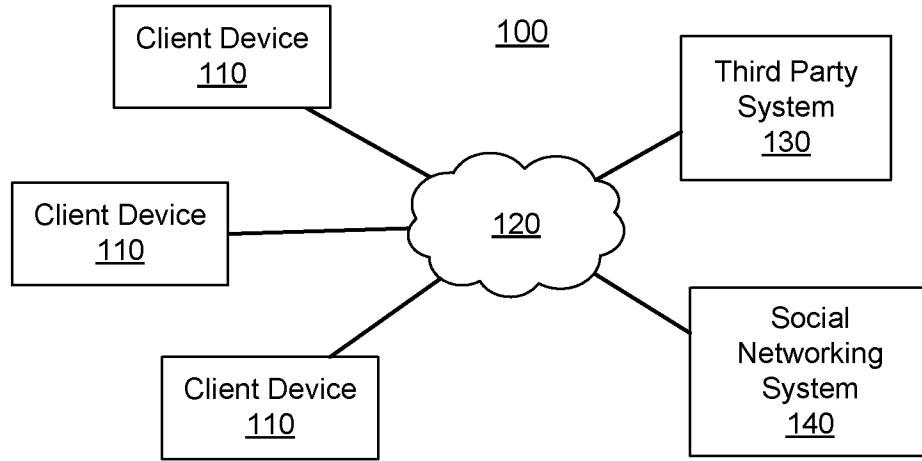


FIG. 1

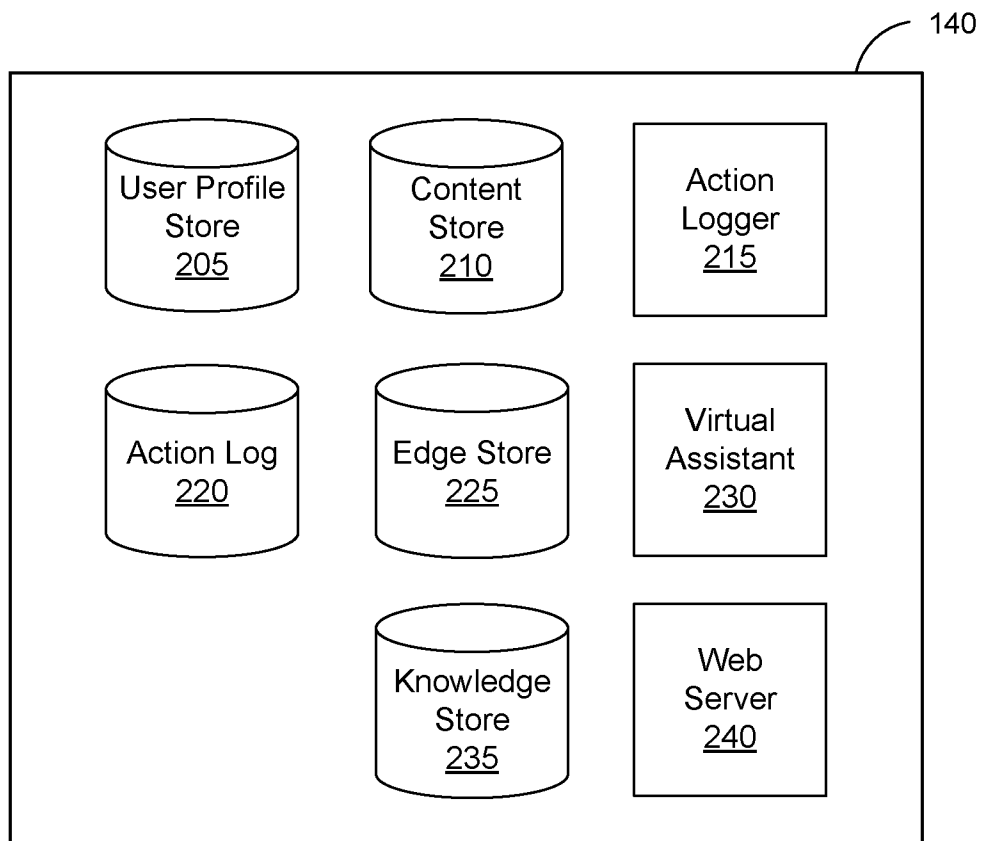


FIG. 2

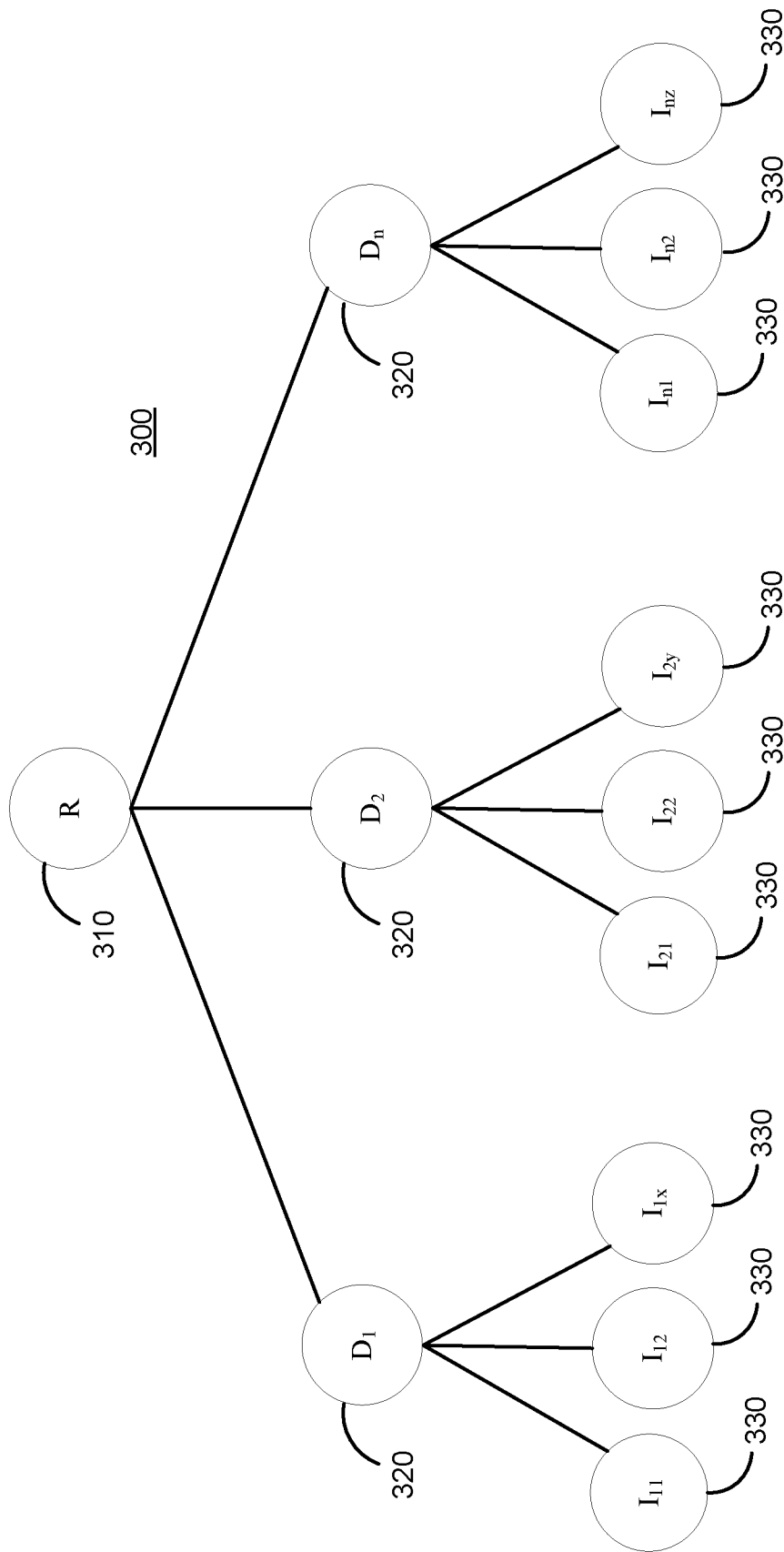


FIG. 3

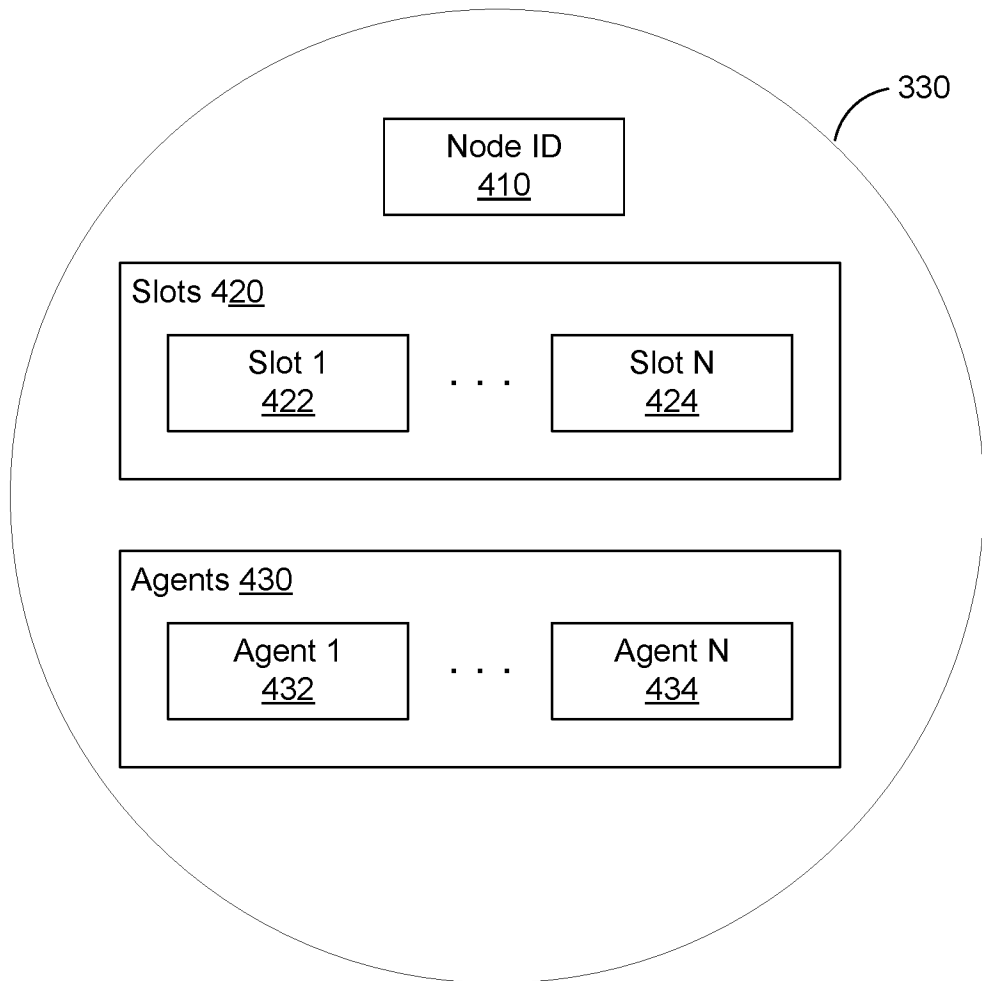


FIG. 4

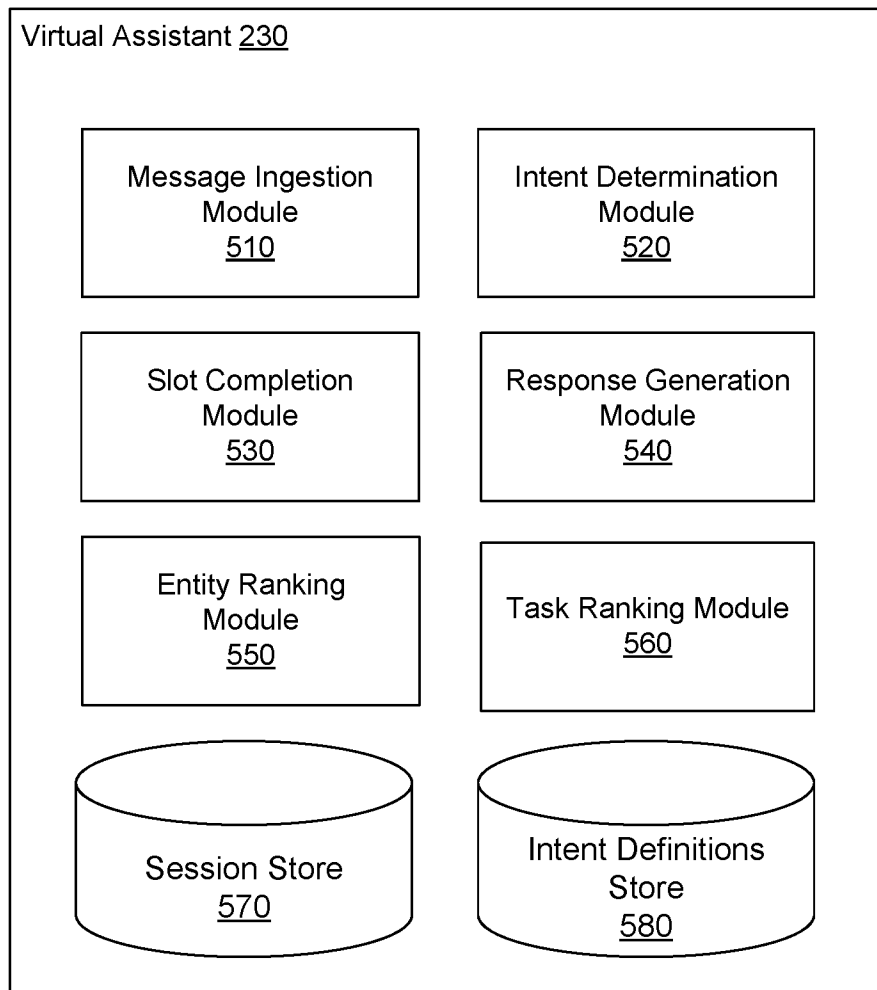


FIG. 5

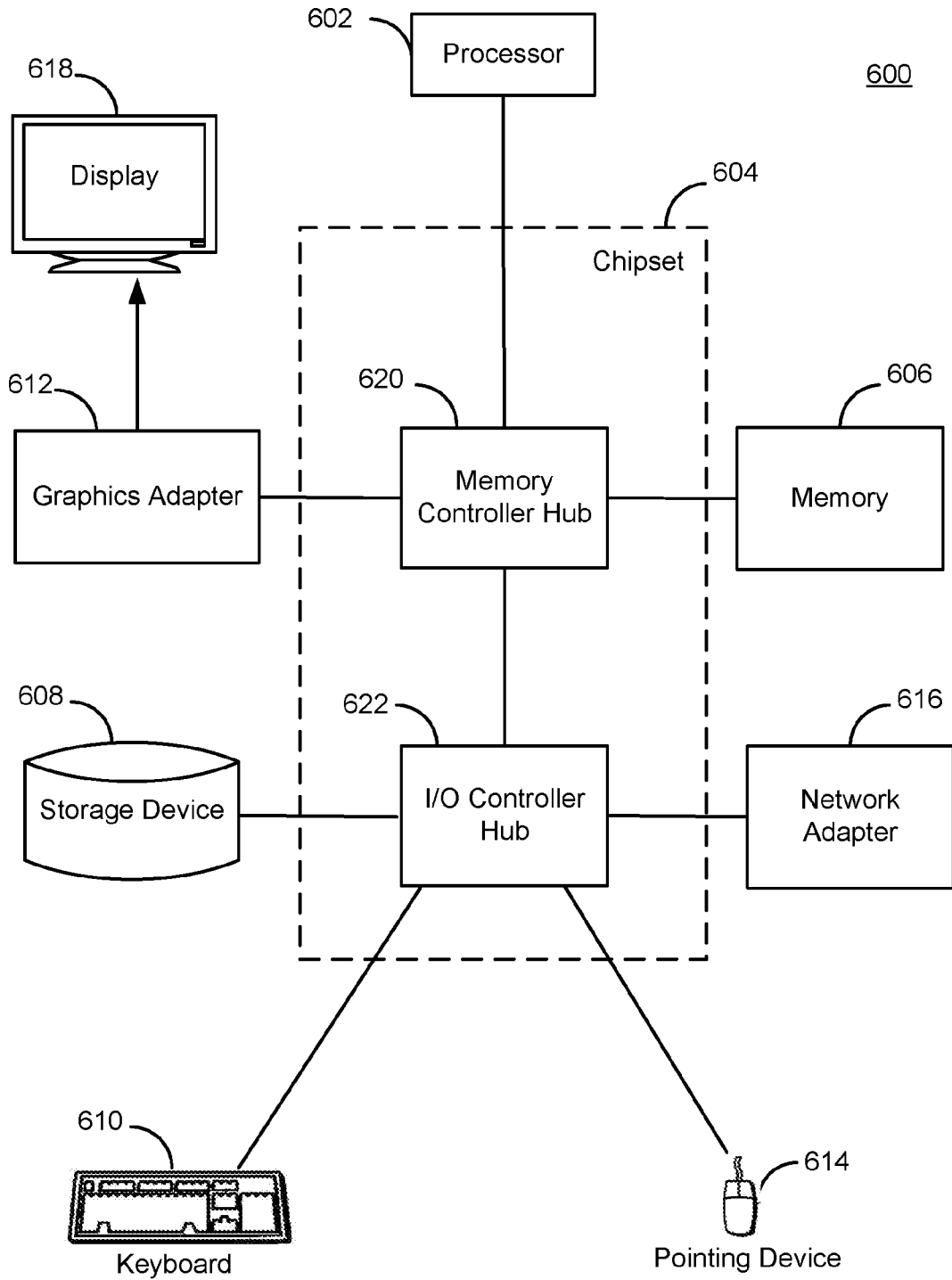


FIG. 6

6/6

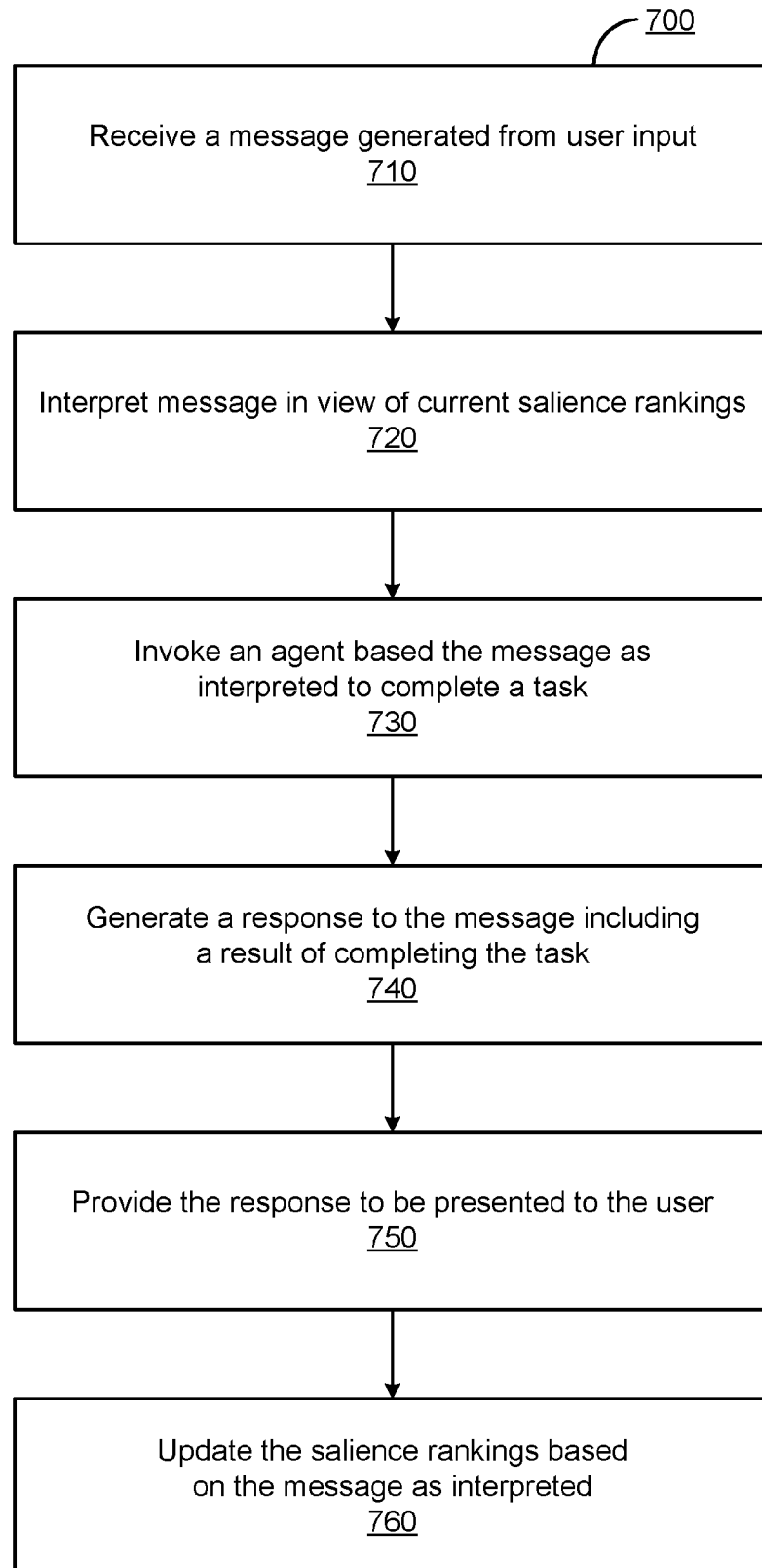


FIG. 7

A. CLASSIFICATION OF SUBJECT MATTER**G06F 16/00(2019.01)i**

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

B. FIELDS SEARCHEDMinimum documentation searched (classification system followed by classification symbols)
G06F 16/00; G06F 17/20; G06F 17/28; G06F 17/30; G10L 15/06; G10L 15/22; G10L 21/00Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility modelsElectronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: message, user, intent, natural language, task, analyze, rank, list, relevance, session, assign, entity, slot, agent, and similar terms.**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2016-0225370 A1 (MICROSOFT TECHNOLOGY LICENSING, LLC) 04 August 2016 See paragraphs [0004]-[0007], [0036], and [0050]-[0060]; claims 1-2; and figures 1-7.	1-20
A	US 2016-0154792 A1 (MICROSOFT TECHNOLOGY LICENSING, LLC) 02 June 2016 See paragraphs [0003] and [0053]-[0070]; and figures 4-6.	1-20
A	US 2014-0040281 A1 (BOTTLENOSE, INC.) 06 February 2014 See paragraphs [0119]-[0127] and figures 16-17.	1-20
A	US 2012-0016678 A1 (THOMAS ROBERT GRUBER et al.) 19 January 2012 See paragraphs [0008]-[0022] and figure 22.	1-20
A	WO 2016-108873 A1 (FACEBOOK, INC. et al.) 07 July 2016 See paragraphs [0002]-[0004] and [0071]-[0083]; and figures 6-7.	1-20

 Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

15 January 2019 (15.01.2019)

Date of mailing of the international search report

16 January 2019 (16.01.2019)

Name and mailing address of the ISA/KR

International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

NHO, Ji Myong

Telephone No. +82-42-481-8528



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date		
US 2016-0225370 A1	04/08/2016	AU 2016-211903 A1	15/06/2017		
		BR 112017011564 A2	02/01/2018		
		CA 2970728 A1	04/08/2016		
		CL 2017001872 A1	09/02/2018		
		CN 107210033 A	26/09/2017		
		EP 3251115 A2	06/12/2017		
		IL 252454 D0	31/07/2017		
		JP 2018-513431 A	24/05/2018		
		KR 10-2017-0115501 A	17/10/2017		
		MX 2017009711 A	17/11/2017		
		PH 12017550013 A1	18/10/2017		
		SG 11201705873 RA	30/08/2017		
		US 9508339 B2	29/11/2016		
		WO 2016-122902 A2	04/08/2016		
		WO 2016-122902 A3	27/10/2016		
		US 2016-0154792 A1	02/06/2016	US 10007660 B2	26/06/2018
				US 2017-262432 A1	14/09/2017
US 9690776 B2	27/06/2017				
US 2014-0040281 A1	06/02/2014	CA 2824605 A1	30/08/2012		
		CA 2824627 A1	30/08/2012		
		CA 2824627 C	30/09/2014		
		CA 2864127 A1	22/08/2013		
		CA 2866615 A1	12/09/2013		
		CA 2866615 C	12/05/2015		
		CA 2879157 A1	06/02/2014		
		CA 2879157 C	05/01/2016		
		CA 2883974 A1	06/02/2014		
		CA 2883974 C	05/01/2016		
		EP 2678784 A2	01/01/2014		
		EP 2678784 A4	06/08/2014		
		EP 2678822 A2	01/01/2014		
		EP 2678822 A4	10/09/2014		
		EP 2747013 A1	25/06/2014		
		EP 2747014 A1	25/06/2014		
		EP 2769292 A1	27/08/2014		
		EP 2769292 A4	07/01/2015		
		EP 2783337 A1	01/10/2014		
		EP 2783337 A4	29/07/2015		
		GB 2502736 A	04/12/2013		
		GB 2502737 A	04/12/2013		
		GB 2507667 A	07/05/2014		
GB 2510546 A	06/08/2014				
GB 2512574 A	08/10/2014				
US 2012-0272160 A1	25/10/2012				
US 2012-0278164 A1	01/11/2012				
US 2012-0296991 A1	22/11/2012				
US 2013-0290317 A1	31/10/2013				

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		US 2013-0298038 A1	07/11/2013
		US 2013-0298084 A1	07/11/2013
		US 2014-0040387 A1	06/02/2014
		US 2014-0129324 A1	08/05/2014
		US 2014-0129331 A1	08/05/2014
		US 2014-0136544 A1	15/05/2014
		US 2014-0136997 A1	15/05/2014
		US 2015-0095021 A1	02/04/2015
		US 2015-0256636 A1	10/09/2015
		US 2016-0094646 A1	31/03/2016
		US 2016-0255037 A1	01/09/2016
		US 2017-0142051 A1	18/05/2017
		US 8832092 B2	09/09/2014
		US 8938450 B2	20/01/2015
		US 8990097 B2	24/03/2015
		US 9009126 B2	14/04/2015
		US 9304989 B2	05/04/2016
		US 9614807 B2	04/04/2017
		US 9876751 B2	23/01/2018
		WO 2012-116236 A2	30/08/2012
		WO 2012-116236 A3	04/04/2013
		WO 2012-116241 A2	30/08/2012
		WO 2012-116241 A3	22/11/2012
		WO 2013-113028 A1	01/08/2013
		WO 2013-123462 A1	22/08/2013
		WO 2013-133966 A1	12/09/2013
		WO 2013-133966 A8	12/09/2013
		WO 2013-133966 A9	12/09/2013
		WO 2014-022538 A1	06/02/2014
		WO 2014-074643 A2	15/05/2014
		WO 2014-074643 A3	03/07/2014
US 2012-0016678 A1	19/01/2012	AU 2009-246654 A1	19/11/2009
		AU 2009-246654 B2	04/10/2012
		AU 2010-254812 A1	12/01/2012
		AU 2011-205426 A1	23/08/2012
		AU 2011-205426 B2	16/01/2014
		AU 2012-101191 A4	25/10/2012
		AU 2012-101191 B4	09/05/2013
		AU 2012-101465 A4	15/11/2012
		AU 2012-101465 B4	23/05/2013
		AU 2012-101466 A4	15/11/2012
		AU 2012-101466 B4	23/05/2013
		AU 2012-101467 A4	22/11/2012
		AU 2012-101467 B4	23/05/2013
		AU 2012-101468 B4	16/05/2013
		AU 2012-101469 A4	15/11/2012
		AU 2012-101469 B4	16/05/2013
		AU 2012-101470 A4	15/11/2012
		AU 2012-101470 B4	16/05/2013

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		AU 2012-101471 A4	15/11/2012
		AU 2012-101471 B4	30/05/2013
		AU 2012-101472 A4	15/11/2012
		AU 2012-101472 B4	16/05/2013
		AU 2012-101473 A4	15/11/2012
		AU 2012-101473 B4	23/05/2013
		AU 2012-232977 A1	18/04/2013
		AU 2012-261958 A1	16/01/2014
		AU 2012-261958 B2	21/04/2016
		AU 2012-316484 A1	17/04/2014
		AU 2013-200021 A1	24/01/2013
		AU 2013-200021 B2	10/03/2016
		AU 2013-205568 A1	16/05/2013
		AU 2013-205568 B2	04/06/2015
		AU 2013-205569 A1	16/05/2013
		AU 2013-205569 B2	25/02/2016
		AU 2013-205571 A1	16/05/2013
		AU 2013-205571 B2	11/06/2015
		AU 2013-205584 A1	16/05/2013
		AU 2013-205584 B2	24/03/2016
		AU 2013-205585 A1	16/05/2013
		AU 2013-205585 B2	26/11/2015
		AU 2013-205586 A1	16/05/2013
		AU 2013-205586 B2	20/08/2015
		AU 2013-205588 A1	16/05/2013
		AU 2013-205588 B2	25/02/2016
		AU 2013-205590 A1	16/05/2013
		AU 2013-205590 B2	04/06/2015
		AU 2013-205591 A1	16/05/2013
		AU 2013-205591 B2	25/02/2016
		AU 2013-260186 A1	04/12/2014
		AU 2013-262803 A1	27/11/2014
		AU 2013-262803 B2	21/07/2016
		AU 2013-344629 A1	04/06/2015
		AU 2013-344629 B2	02/03/2017
		AU 2014-100584 A4	03/07/2014
		AU 2014-100584 B4	05/03/2015
		AU 2014-100585 A4	03/07/2014
		AU 2014-100585 B4	12/03/2015
		AU 2014-221287 A1	25/09/2014
		AU 2014-221287 B2	14/07/2016
		AU 2014-235244 A1	24/09/2015
		AU 2014-235244 B2	13/04/2017
		AU 2014-235245 A1	24/09/2015
		AU 2014-235245 B2	02/03/2017
		AU 2014-235246 A1	24/09/2015
		AU 2014-235246 B2	17/08/2017
		AU 2014-235248 A1	24/09/2015
		AU 2014-235248 B2	11/05/2017
		AU 2014-274913 A1	12/11/2015

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		AU 2014-274913 B2	11/05/2017
		AU 2014-275224 A1	24/12/2015
		AU 2014-275224 B2	31/08/2017
		AU 2015-100708 A4	09/07/2015
		AU 2015-100708 B4	03/03/2016
		AU 2015-100709 A4	09/07/2015
		AU 2015-100709 B4	10/03/2016
		AU 2015-101019 A4	10/09/2015
		AU 2015-101020 A4	10/09/2015
		AU 2015-101020 B4	21/04/2016
		AU 2015-101021 A4	10/09/2015
		AU 2015-101021 B4	19/05/2016
		AU 2015-101022 A4	10/09/2015
		AU 2015-101022 B4	19/11/2015
		AU 2015-101023 A4	10/09/2015
		AU 2015-101023 B4	05/11/2015
		AU 2015-101183 A4	08/10/2015
		AU 2015-101183 B4	19/11/2015
		AU 2015-101188 A4	08/10/2015
		AU 2015-101188 B4	20/10/2016
		AU 2015-203483 A1	16/07/2015
		AU 2015-214079 A1	18/08/2016
		AU 2015-214079 B2	08/09/2016
		AU 2015-214079 C1	19/01/2017
		AU 2015-217268 A1	21/07/2016
		AU 2015-217268 B2	01/03/2018
		AU 2015-266650 A1	13/10/2016
		AU 2015-266650 B2	01/02/2018
		AU 2015-266693 A1	06/10/2016
		AU 2015-266693 B2	01/02/2018
		AU 2015-298710 A1	09/02/2017
		AU 2015-312215 A1	06/10/2016
		AU 2015-312215 B2	19/10/2017
		AU 2015-312369 A1	23/03/2017
		AU 2015-312369 B2	05/04/2018
		AU 2016-100090 A4	10/03/2016
		AU 2016-100090 B4	27/10/2016
		AU 2016-100367 A4	05/05/2016
		AU 2016-100367 B4	24/11/2016
		AU 2016-100383 A4	05/12/2016
		AU 2016-100383 B4	22/12/2016
		AU 2016-100383 C4	17/05/2018
		AU 2016-100411 A4	19/05/2016
		AU 2016-100476 A4	26/05/2016
		AU 2016-100476 B4	22/12/2016
		AU 2016-100765 A4	23/06/2016
		AU 2016-102002 A4	15/12/2016
		AU 2016-102002 B4	19/01/2017
		AU 2016-102031 A4	22/12/2016
		AU 2016-102031 B4	24/08/2017

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		AU 2016-200568 A1	18/02/2016
		AU 2016-204091 A1	07/07/2016
		AU 2016-204091 B2	15/03/2018
		AU 2016-204262 A1	14/07/2016
		AU 2016-204262 B2	18/01/2018
		AU 2016-204683 A1	21/07/2016
		AU 2016-204683 B2	25/01/2018
		AU 2016-204980 A1	16/03/2017
		AU 2016-211504 A1	21/09/2017
		AU 2016-211504 B2	26/10/2017
		AU 2016-215440 A1	17/08/2017
		AU 2016-229847 A1	21/09/2017
		AU 2016-244329 A1	03/11/2016
		AU 2017-100070 A4	02/03/2017
		AU 2017-100070 B4	28/09/2017
		AU 2017-100197 A4	23/03/2017
		AU 2017-100197 B4	26/10/2017
		AU 2017-100198 A4	23/03/2017
		AU 2017-100198 B4	26/10/2017
		AU 2017-201064 A1	09/03/2017
		AU 2017-201064 B2	01/03/2018
		AU 2017-201068 A1	09/03/2017
		AU 2017-203847 A1	22/06/2017
		AU 2017-204217 A1	13/07/2017
		AU 2017-261594 A1	07/12/2017
		AU 2017-268569 A1	21/12/2017
		AU 2018-200428 A1	08/02/2018
		AU 2018-200628 A1	15/02/2018
		AU 2018-201089 A1	08/03/2018
		AU 2018-202411 A1	26/04/2018
		AU 2018-202559 A1	10/05/2018
		BR 102012024861 A2	14/02/2017
		BR 112012017826 A2	26/09/2017
		BR 122012028965 A2	27/03/2018
		BR PI0912741 A2	13/10/2015
		CA 2787351 A1	21/07/2011
		CA 2787351 C	08/12/2015
		CA 2791277 A1	30/03/2013
		CA 2791791 A1	21/07/2011
		CA 2791791 C	18/10/2016
		CA 2792412 A1	21/07/2011
		CA 2792412 C	22/12/2015
		CA 2792442 A1	21/07/2011
		CA 2792442 C	17/05/2016
		CA 2792570 A1	21/07/2011
		CA 2792570 C	31/05/2016
		CA 2793002 A1	21/07/2011
		CA 2793002 C	21/06/2016
		CA 2793118 A1	21/07/2011
		CA 2793118 C	08/08/2017

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		CA 2793248 A1	21/07/2011
		CA 2793248 C	01/11/2016
		CA 2793741 A1	21/07/2011
		CA 2793741 C	31/05/2016
		CA 2793743 A1	21/07/2011
		CA 2793743 C	06/10/2015
		CA 2954559 A1	21/07/2011
		CA 3000109 A1	21/07/2011
		CN 101582053 A	18/11/2009
		CN 101582053 B	22/01/2014
		CN 102792320 A	21/11/2012
		CN 102792320 B	24/02/2016
		CN 103226949 A	31/07/2013
		CN 103226949 B	01/02/2017
		CN 103582896 A	12/02/2014
		CN 103778082 A	07/05/2014
		CN 103778082 B	05/04/2017
		CN 103959751 A	30/07/2014
		CN 104335205 A	04/02/2015
		CN 104335205 B	19/10/2018
		CN 104395860 A	04/03/2015
		CN 104412201 A	11/03/2015
		CN 104685470 A	03/06/2015
		CN 104834419 A	12/08/2015
		CN 104937553 A	23/09/2015
		CN 105051494 A	11/11/2015
		CN 105051494 B	26/01/2018
		CN 105051495 A	11/11/2015
		CN 105051496 A	11/11/2015
		CN 105143828 A	09/12/2015
		CN 105191387 A	23/12/2015
		CN 105247511 A	13/01/2016
		CN 105283840 A	27/01/2016
		CN 105284099 A	27/01/2016
		CN 105303372 A	03/02/2016
		CN 105320454 A	10/02/2016
		CN 105320455 A	10/02/2016
		CN 105321067 A	10/02/2016
		CN 105335087 A	17/02/2016
		CN 105379234 A	02/03/2016
		CN 105388998 A	09/03/2016
		CN 105389078 A	09/03/2016
		CN 105487790 A	13/04/2016
		CN 105718185 A	29/06/2016
		CN 105808200 A	27/07/2016
		CN 105844462 A	10/08/2016
		CN 105981352 A	28/09/2016
		CN 105981352 B	13/03/2018
		CN 106068490 A	02/11/2016
		CN 106462123 A	22/02/2017

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		CN 106537397 A	22/03/2017
		CN 106664226 A	10/05/2017
		CN 106797333 A	31/05/2017
		CN 106797493 A	31/05/2017
		CN 106961334 A	18/07/2017
		CN 107209655 A	26/09/2017
		CN 107430489 A	01/12/2017
		CN 107633397 A	26/01/2018
		CN 107667337 A	06/02/2018
		CN 107683439 A	09/02/2018
		CN 107683601 A	09/02/2018
		CN 107851046 A	27/03/2018
		CN 107921317 A	17/04/2018
		CN 204650490 U	16/09/2015
		CN 204650596 U	16/09/2015
		CN 205158436 U	13/04/2016
		CN 205334368 U	22/06/2016
		CN 205486013 U	17/08/2016
		CN 205608658 U	28/09/2016
		CN 206193905 U	24/05/2017
		CN 206193906 U	24/05/2017
		CN 206649467 U	17/11/2017
		DE 102012019178 A1	04/04/2013
		DE 112013002410 T5	22/01/2015
		DE 112013002421 T5	05/02/2015
		DE 112015002326 T5	23/03/2017
		DE 112015003083 T5	11/05/2017
		DE 202014004555 U1	18/09/2014
		DE 202014004560 U1	12/09/2014
		DE 202015004267 U1	11/12/2015
		DE 202015004268 U1	13/11/2015
		DE 202015005394 U1	08/12/2015
		DE 202015005395 U1	17/11/2015
		DE 202015005397 U1	08/12/2015
		DE 202015005399 U1	12/11/2015
		DE 202015005400 U1	08/12/2015
		DE 202015006055 U1	02/02/2016
		DE 202015006066 U1	14/12/2015
		DE 202015009088 U1	06/09/2016
		DE 202015009089 U1	09/09/2016
		DE 202016001214 U1	23/06/2016
		DK 178589 B1	01/08/2016
		DK 178595 B1	08/08/2016
		DK 178771 B1	09/01/2017
		DK 178931 B1	12/06/2017
		DK 178964 B1	10/07/2017
		DK 179222 B1	12/02/2018
		DK 179348 B1	14/05/2018
		DK 201570495 A1	29/02/2016
		DK 201570496 A1	18/07/2016

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		DK 201570497 A1	29/02/2016
		DK 201570498 A1	22/02/2016
		DK 201570499 A1	29/02/2016
		DK 201570550 A1	14/03/2016
		DK 201570563 A1	14/03/2016
		DK 201570666 A1	25/07/2016
		DK 201570668 A1	25/07/2016
		DK 201570775 A1	09/01/2017
		DK 201670042 A1	22/08/2016
		DK 201670319 A1	30/05/2016
		DK 201670320 A1	06/06/2016
		DK 201670656 A1	19/09/2016
		DK 201770125 A1	13/03/2017
		DK 201770126 A1	20/03/2017
		DK 201770191 A1	27/03/2017
		DK 201770292 A1	08/05/2017
		EP 2283424 A2	16/02/2011
		EP 2526511 A2	28/11/2012
		EP 2575128 A2	03/04/2013
		EP 2575128 A3	14/08/2013
		EP 2715625 A2	09/04/2014
		EP 2761860 A1	06/08/2014
		EP 2778614 A1	17/09/2014
		EP 2778614 B1	20/07/2016
		EP 2778615 A2	17/09/2014
		EP 2778615 A3	29/07/2015
		EP 2847693 A2	18/03/2015
		EP 2920693 A1	23/09/2015
		EP 2946171 A2	25/11/2015
		EP 2946172 A1	25/11/2015
		EP 2972104 A1	20/01/2016
		EP 2992418 A1	09/03/2016
		EP 3005075 A1	13/04/2016
		EP 3005150 A2	13/04/2016
		EP 3005668 A1	13/04/2016
		EP 3072040 A1	28/09/2016
		EP 3101392 A1	07/12/2016
		EP 3103246 A1	14/12/2016
		EP 3103246 B1	06/12/2017
		EP 3108426 A1	28/12/2016
		EP 3108427 A1	28/12/2016
		EP 3129907 A2	15/02/2017
		EP 3131023 A1	15/02/2017
		EP 3149547 A1	05/04/2017
		EP 3149548 A2	05/04/2017
		EP 3154014 A1	12/04/2017
		EP 3156958 A1	19/04/2017
		EP 3158425 A1	26/04/2017
		EP 3175344 A1	07/06/2017
		EP 3176742 A1	07/06/2017

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		EP 3189409 A1	12/07/2017
		EP 3189416 A2	12/07/2017
		EP 3190747 A1	12/07/2017
		EP 3195096 A1	26/07/2017
		EP 3200185 A1	02/08/2017
		EP 3201773 A1	09/08/2017
		EP 3251068 A1	06/12/2017
		EP 3254452 A1	13/12/2017
		EP 3268844 A1	17/01/2018
		EP 3281388 A1	14/02/2018
		EP 3286642 A1	28/02/2018
		EP 3286702 A1	28/02/2018
		EP 3313050 A1	25/04/2018
		GB 2459956 A	18/11/2009
		GB 2459956 B	25/08/2010
		GB 2472482 A	09/02/2011
		GB 2472482 B	21/09/2011
		GB 2490444 A	31/10/2012
		GB 2495222 A	03/04/2013
		GB 2495222 B	26/10/2016
		HK 1137831 A1	14/01/2011
		HK 1200621 A1	07/08/2015
		HK 1205312 A1	11/12/2015
		HK 1206835 A1	15/01/2016
		HK 1212486 A1	10/06/2016
		HK 1219144 A1	24/03/2017
		HK 1220023 A1	21/04/2017
		HK 1220525 A1	05/05/2017
		HK 1220535 A1	05/05/2017
		HK 1220536 A1	05/05/2017
		HK 1221037 A1	19/05/2017
		HK 1221038 A1	19/05/2017
		HK 1221039 A1	19/05/2017
		HK 1222726 A1	07/07/2017
		HK 1222922 A1	14/07/2017
		HK 1222923 A1	14/07/2017
		HK 1223694 A1	04/08/2017
		HK 1224110 A1	11/08/2017
		JP 2013-080476 A	02/05/2013
		JP 2013-517566 A	16/05/2013
		JP 2014-222509 A	27/11/2014
		JP 2014-222510 A	27/11/2014
		JP 2014-222511 A	27/11/2014
		JP 2014-222512 A	27/11/2014
		JP 2014-222513 A	27/11/2014
		JP 2014-222514 A	27/11/2014
		JP 2014-222515 A	27/11/2014
		JP 2014-222516 A	27/11/2014
		JP 2014-222517 A	27/11/2014
		JP 2014-518409 A	28/07/2014

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/030390

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		JP 2015-122104 A	02/07/2015
		JP 2015-501022 A	08/01/2015
		JP 2015-519838 A	09/07/2015
		JP 2017-016683 A	19/01/2017
		JP 2017-139004 A	10/08/2017
		JP 2017-142833 A	17/08/2017
		JP 2017-224300 A	21/12/2017
		JP 2018-116728 A	26/07/2018
		JP 5137899 B2	06/02/2013
		JP 5698203 B2	08/04/2015
		JP 5948372 B2	06/07/2016
		JP 5956511 B2	27/07/2016
		JP 5957038 B2	27/07/2016
		JP 5973500 B2	23/08/2016
		JP 6027052 B2	16/11/2016
		JP 6144256 B2	07/06/2017
		JP 6175413 B2	02/08/2017
		JP 6193181 B2	06/09/2017
		JP 6193361 B2	06/09/2017
		JP 6285883 B2	28/02/2018
		JP 6353711 B2	04/07/2018
		JP 6353786 B2	04/07/2018
		JP 6356501 B2	11/07/2018
WO 2016-108873 A1	07/07/2016	AU 2014-415625 A1	06/07/2017
		CA 2970242 A1	07/07/2016
		IL 252798 D0	31/08/2017
		JP 2018-506780 A	08/03/2018
		KR 10-2017-0099969 A	01/09/2017
		US 10095687 B2	09/10/2018
		US 2016-0188571 A1	30/06/2016
		US 2017-132211 A1	11/05/2017
		US 2017-132212 A1	11/05/2017
		US 2018-032509 A1	01/02/2018
		US 9589305 B2	07/03/2017
		US 9817815 B2	14/11/2017
		US 9817816 B2	14/11/2017