October 02, 2017

DISAMBIGUATING NOTES OF DIFFERENT USERS IN A MULTI-USER ASSISTANT ENVIRONMENT

Google Inc.

DISAMBIGUATING NOTES OF DIFFERENT USERS IN A MULTI-USER ASSISTANT ENVIRONMENT

ABSTRACT

A virtual, intelligent, or computational assistant (e.g., also referred to simply as an "assistant") is described that stores and organizes user created information (e.g., memos, notes, lists, reminders, etc.) on behalf of multiple users of the assistant. The assistant annotates, tags, or otherwise organizes user created information according to which particular user the information was created for. This way, during a future interaction between a particular user and the assistant, the assistant can easily identify and retrieve the information that is associated with that particular user and not some other user of the assistant.

DESCRIPTION

Assistants (e.g., virtual, intelligent, or computational assistants) execute on counter-top computing devices, mobile phones, automobiles, and many other types of computing devices. Assistants output useful information, responds to users' needs, or otherwise performs certain operations to help users complete real-world and/or virtual tasks. Some assistants are configured to record reminders, notes, lists, and other information as individualized, persistent records that each user can later retrieve or modify during subsequent interactions with the assistant. Some assistants are configured to handle multiple users. While providing a user interface for creating persistent records for individual users may be somewhat trivial, organizing and maintaining multiple, persistent records on behalf of multiple users may be challenging for some assistants.

In some cases, different users of a single assistant may wish to create or later retrieve similar sounding information, even though each user may be referring to a different piece of

information being maintained by the assistant. For example, two different users may interact with a single assistant to create their own, unique, personalized shopping lists. A first user who intends to go shopping at a grocery store, may tell the assistant to create a shopping list that includes apples. The assistant may determine the identity of the first user and tag the shopping list with a user identifier indicative of the first user so that the shopping list with the apples is forever associated with the first user, i.e., the particular user that was interacting with the assistant when the assistant created or last modified that shopping list. A second user who intends to go shopping at a members-only warehouse store may tell the assistant to create a shopping list that includes printer paper. The assistant may determine the identity of the second user and tag the shopping list with a user identifier indicative of the second user so that the shopping list with the printer paper is linked to the second user.

Later the first user may tell the assistant to add milk to the shopping list and the second user may tell the assistant to add oranges to the shopping list. Because the assistant maintains multiple shopping lists for multiple users, each of the first and second users' instructions referencing "shopping list" are ambiguous. But because the first and second users may be unaware that the assistant maintains the multiple shopping lists, the first and second users may not know the need, or even how, direct the assistant to the correct list.

The example system shown in FIG. 1 provides an assistant architecture that automatically tags user created information with supplemental data (e.g., user identifiers) indicative of which particular user the information was created for. Before the assistant processes a user instruction to handle user created information, the assistant first identifies the intended user for the instruction and then ensures that the user created information the assistant touches to process the user instruction, matches the intended user.

For example, if instead simply processing an instruction from the first user to add milk to "the shopping list" the assistant first identifies the first user as the particular user currently instructing the assistant to do something. Then, instead of necessarily doing a global search in memory for "shopping lists", the assistant searches its information stores for a shopping list tagged with a user identifier of the first user. That is, the assistant matches supplemental data attached to user created information with the identity of the user providing the instruction. The assistant therefore modifies the shopping list created by the first user (i.e., the shopping list that includes apples), to include milk, rather than modifying the shipping list created by the second user (i.e., the shopping list that includes printer paper).

Further to the descriptions below, a user may be provided with controls allowing the user to make an election as to both if and when the assistant, the computing device, or the computing systems described herein can collect or make use of user information (e.g., user information or contextual information about a user's social network, social actions or activities, profession, a user's preferences, or a user's current location), and if and when the user is sent content or communications from a server. In addition, certain data may be treated in one or more ways before it is stored or used, so that personally identifiable information is removed. For example, a user's identity may be treated so that no personally identifiable information can be determined for the user, or a user's geographic location may be generalized where location information is obtained (such as to a city, ZIP code, or state level), so that a particular location of a user cannot be determined. Thus, the user may have control over what user information is collected about the user, how that user information is used, and what user information is provided to the user.

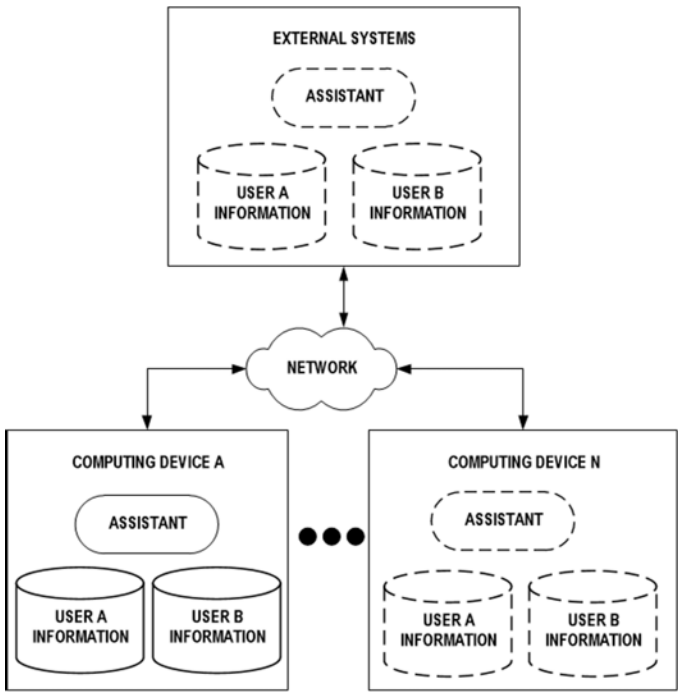

FIG. 1

The system of FIG. 1 includes one or more external systems and computing devices A–N communicating across a network to provide an assistant service that maintains user information. The network of FIG. 1 represents a combination of any one or more public or private communication networks, for instance, television broadcast networks, cable or satellite networks, cellular networks, Wi-Fi networks, broadband networks, and/or other type of network for transmitting data (e.g., telecommunications and/or media data) between various computing devices, systems, and other communications and media equipment.

Computing devices A–N represent any type of computing device that is configured to execute an assistant and communicate on a network. The external systems represent any type of server or other computing system that is configured to support the assistants executing at computing devices A–N.

The external systems and computing devices A–N can be personal, corporate, or government owned computing devices. In some examples, computing devices A–N may be

shared assets of multiple users. Examples of computing devices A–N include servers, mainframes, cloud computing environments, mobile phones, tablet computers, wearable computing devices, countertop computing devices, home automation computing devices, laptop computers, desktop computers, televisions, stereos, automobiles, and any and all other type of mobile and non-mobile computing device that is configured to execute an assistant. For example, computing device A may be a countertop assistant device and computing device N may be a mobile phone or automobile infotainment system.

An Assistant executes across the external systems and one or more of computing devices A–N to provide assistant services to users of computing devices A–N. Examples of assistant services include: setting up reminders, creating calendar entries, booking travel, online ordering, sending messages or other communications, controlling televisions, lights, thermostats, appliances, or other computing devices, providing navigational instructions, or any other conceivable task or operation that may be performed by an assistant.

As a user interacts with the assistant, the assistant may obtain personal information about the user. Examples of personal information include: habits, preferences, notes, lists, contacts, communications, interests, location histories, and other types of user information. After receiving explicit permission from the user, the assistant may store, the personal information at user information data stores and in the course of providing assistant services, make use of the personal information stored at the user information data stores.

The external systems, computing devices A–N, and the assistant treat the information stored at the information stores so that the information is protected, encrypted, or otherwise not susceptible to hacking or unauthorized use. The information stored at the information data stores may be stored locally at each of computing devices A–N and/or remotely (e.g., in a cloud

computing environment provided by the external systems and which is accessible via the network of FIG. 1).

The assistant stores and organizes user created information (e.g., memos, notes, lists, reminders, etc.) on behalf of multiple users of the assistant. Of course, in order to organize and later retrieve user created information that is associated with a particular user, the assistant needs to determine the identity of the user currently interacting with the assistant.

The assistant can determine a user identity in one of many ways. The assistant may require the user to self-identify. For instance, before performing any operations on behalf of a user, the assistant may require the user to provide a credential (e.g., a user name, a password, etc.) to the assistant or otherwise provide an input to the assistant that lets the assistant know which user is currently interacting with the assistant. In cases where the assistant provides a voice interface, the assistant may identify a user using voice recognition or voiceprint technology. The assistant may infer the identity of a user using biometrics (e.g., facial recognition, fingerprint, etc.) or any other technique. In some cases, the assistant can determine the identity of a user by attributing the user identity to the identity of a user associated with a secondary device that is in proximity to the assistant when the user is communicating with the assistant. For example, when the assistant receives input from a user the assistant may also detect a radio frequency identifier tag or a Bluetooth signal emanating from a nearby device and infer that the user providing the input is the user associated with the nearby device.

The assistant annotates, tags, or otherwise organizes user created information according to which particular user the information was created. For example, as shown in FIG. 1, computing devices A–N retain information created by each of users A and B in separate, and individual data stores. This way, during a future interaction with user A, the assistant can easily

identify and retrieve the information that is associated with user A and not user B or some other user of the assistant. In other examples, the assistant may store user created information associated with multiple users in a single data store and annotate or tag each piece of information stored at the information data stores with supplemental data that indicates which of the multiple users the information is created. In other words, rather than include a separate individual data store for each user (as shown in FIG. 1), the assistant may store all user created information together, but annotate each separate piece of information with a user identifier so the assistant can later retrieve the information that is particular to a specific user.

In some cases, users may wish to share information with other users and the assistant may enable the users to reference the information. For example, user A may create user information that the assistant annotates as being associated with user A. User A want to share the information with user B, and therefore, the assistant may annotate the user information further as being shared by user A with user B. This way, user B may interact with the assistant by commanding the assistant to provide “information shared with me” or “information shared with me by user A” or even “information created by user A”.

In a first example, computing device A is a counter-top assistant shared in the home of user A and user B. Before leaving for work, user A may interact with computing device A by telling the assistant “remind me to take out the garbage tonight”. The assistant may determine using voice-print technology that user A provided the assistant with the command to create the reminder and in response, the assistant may generate the reminder and tag the reminder with a user identifier of user A. Later that day, user B may interact with computing device A telling the assistant “remind me to pack my rain gear for my camping trip”. The assistant may determine using voice-print technology that user B provided the assistant with the command to create the

reminder and in response, the assistant may generate the reminder and tag the reminder with a user identifier of user B.

When user A returns home, user A may ask the assistant executing at computing device A, “what was I suppose to remember to do”. The assistant may determine using voice-print technology that user A provided the assistant with the command to reproduce his or her reminders for that day and in response, the assistant may search its stored reminders for reminders that are tagged with a user identifier of user A. The assistant may identify the reminder to take out the garbage as being associated with user A and in response, output a notification for user A to remember to take out the garbage.

In a second example, after outputting the notification for user A to remember to take out the garbage, user A may tell the assistant “I don’t have time for that – please remind user B to take out the garbage”. The assistant may determine using voice-print technology that user AB provided the assistant with the command to create the reminder but given the content of the command, the assistant may determine the reminder is for user B. In response, the assistant may generate a reminder “user A wanted to remind you to take out the garbage” and tag the reminder with a user identifier of user B.

Lastly, when user B returns home, user B may ask the assistant executing at computing device A, “what was I suppose to remember to do”. The assistant may determine using voice-print technology that user B provided the assistant with the command to reproduce his or her reminders for that day and in response, the assistant may search its stored reminders for reminders that are tagged with a user identifier of user B. The assistant may identify the reminder left by user A “to take out the garbage” as being associated with user B as well as the reminder “to pack rain gear”. In response, the assistant may output a notification for user B to

remember to pack rain gear as well as a reminder that user A wanted him or her to take out the garbage".

By organizing, annotating, or otherwise supplementing information it stores on behalf of multiple users in this way, the assistant makes retrieving information that is particular to a specific user easier both for the assistant and the user. The above examples are just some use cases for the assistant architecture shown in FIG. 1, the assistant architecture has many other applications and use cases.