Technical Disclosure Commons

Defensive Publications Series

July 03, 2017

Pooling Engine

John D. Lanza
Foley & Lardner LLP

John D. Lanza
Foley & Lardner LLP

POOLING ENGINE

ABSTRACT

Systems and methods described herein allow for consolidation of requests from multiple users in a voice activated interaction computer environment. A data processing system can receive at least two audio input signals from at least two user devices associated with at least two corresponding users, and determine, for each received input audio signal, a respective user request. The data processing system can generate, for each received input audio signal, a corresponding action data structure, and compare the action data structures associated with separate user requests to determine a pooling parameter indicative of overlapping features or themes among the action data structures. The data processing system can generate a pooled action data structure based on the determined pooling parameter, and transmit the pooled data structure to computing device associated with a service provider.

DETAILED DESCRIPTION

In a voice activated computer environment, users can engage in audio conversations with a virtual personal assistant to request services. For instance, users can make audio requests for cab rides, plane tickets, streaming services, restaurants, movies, electronic shopping, etc. A data processing system can process input audio signals received from client devices to determine and handle corresponding user requests. In particular, the data processing system can provide determined user requests to service providers for fulfilling such user requests. The number of user requests handled by the data processing system per unit time can be large. Also, many requests handled by the data processing system at any point of time can be redundant, for example, with respect to the services requested, the corresponding service providers, timing or location parameters related to the requested services, other service features, or a combination thereof.

Systems and methods of the present disclosure relate generally to a data processing system that manages redundant or overlapping audio user requests in a voice activated computer network environment. Each audio user request can be indicative of one or more actions to be executed by service provider or computing device thereof. The data processing system can merge redundant or overlapping requests by batching multiple common actions among multiple user requests into a single (or fewer) corresponding actions. The data processing system can then transmit the batched or pooled action(s) to the corresponding service provider instead of transmitting the individual redundant actions for individual user requests.

The merging or pooling of redundant actions associated with separate user requests can allow for reduced processor utilization, power consumption, bandwidth utilization, and memory usage compared to transmitting the redundant actions individually to respective service providers. In particular, the merging or pooling of redundant actions leads to reduced communications between the data processing system and computing devices of service providers. In addition, by merging or pooling of redundant actions, the data processing system can accommodate service features that are by their nature susceptible to pooling. For example, many users may be willing to share a requested service (e.g., rides or shipping of products) with other users to reduce cost. The merging and pooling of redundant actions allow the data processing system to accommodate such user preferences.

FIG. 1 is a flowchart depicting an example method 100 for managing online actions triggered by multiple users in a voice activated computer network environment. The method 100 can include, at step 105, receiving at least two audio input signals from at least two user devices associated with at least two corresponding users. The users can independently initiate conversations with a virtual personal assistant via their client devices. Each user can make an audio request for an online service (e.g., a ride service, a plane ticket, an online streaming session, online shopping, etc.) or an online search by providing an audio search query. Each client device can forward the respective input audio signal received from the corresponding user to a data processing system hosting the virtual personal assistant.

At step 110, the data processing can determine, for each received input audio signal, a respective user request indicative of a service or online action requested by the corresponding user. The data processing system can machine-translate each of the received input audio signals to a corresponding text. A natural language processor (NLP) component of the data processing system can parse each generated text to identify one or more respective keywords, and identify a user request based on the identified keywords. For example, a generated text including "I need a ride home," the NLP component can identify the keywords "ride," and identify a user request for a ride. The NLP component can also identify the keyword "home" and determine that the ride destination is the user's home.

At step 115, the data processing system can generate, for each received input audio signal, a corresponding action data structure based on the user request and/or keywords identified from that input audio signal. Each action data structure can include information related to a respective user request. For example, for a user request for a cab ride, the corresponding action data structure can include a ride request template generated by the data processing system for sending to a computing device of a service provider. If a user request is related to searching or booking a flight, the corresponding action data structure can include various search queries specifying parameters of the requested flight (e.g., departure location, arrival location, date(s) of flight, etc.) to be submitted to a travel website or a search engine. For a given user request, the data processing system may generate multiple action data structures corresponding to multiple online actions to be performed or may generate a multi-task action data structure.

At step 120, the data processing system can compare the action data structures associated with separate user requests to determine a pooling parameter. The pooling parameter can be indicative of a common requested service, a common service provider, a common requested feature, or other common parameters among two or more action data structures associated with various user requests from separate users. In particular, the data processing system can analyze and compare the generated action data structures to identify overlapping themes, common features, or similarities among various action data structures. For example, the data processing system can parse the action data structures and compare respective content or parameters. The data processing system may further use other information (e.g., geographical map data, user profile data, device information data, etc.) to identify similarities between action data structures. The data processing system can determine the pooling parameter based on identified similarities.

For example, the data processing system may determine, by analyzing and comparing the generated action data structures associated with different users, that multiple users requested a ride and indicated in their requests that they are willing to share a ride with others. The data processing system can then use geographical map data to compare ride routes based on pick-up locations and destinations specified in various action data structures. The data processing system may determine that two or more users are eligible for sharing a ride based on, for example, overlap in respective ride routes. The data processing system can, in response, define a pooling parameter indicative of common intent of sharing a ride and the overlap in (or proximity between) ride routes.

In another example, the data processing system may determine that multiple action data structures relate to (or share a common query for) searching for airplane tickets for a flight between a common departure location and a common arrival location. The data processing system may define the pooling parameter as the flight between the specified departure location and the specified arrival location. In yet another example, the data processing system may determine that the action data structures relate to requests to purchase a service or product from a common product or service provider. In such case, the data processing system may determine the pooling parameter to be the common product or service provider. In general, the pooling parameter can be related to a common service or product, a common service or product provider, a common time parameter, a common geographical parameter, a common service option, or any other common parameter among the action data structures.

At step 125, the data processing system can generate a pooled action data structure based on the determined pooling parameter. For example, the pooled action data structure can be indicative of a ride to be shared by two (or more) users, and can include a first pick-up location, a second pick-up location, a first drop-off location, and a second drop-off location. In another example, the pooled action data structure can include a search query common to multiple user requests made by multiple users. The pooled action data structure may include a coupon provided by a common service or product provider and indication of the service or product requested by two or more users. The coupon may relate to a discount in case of a number of transactions or a transaction amount is reached. The data processing system may obtain the coupon from a website or database associated with the service provider, or some online data source. By grouping two or more requests in the pooled action data structure, the data processing system allows for use of the coupon.

At step 130, the data processing system can transmit the pooled data structure to computing device associated with a service provider. The computing device can process the pooled action data structure and execute actions indicated therein. The computing device can then send a response to the data processing system. The response can be indicative of a confirmation of a requested service, a list of search results, or a request for additional information. The data processing system can transmit the response to the client devices of the users who made the requests consolidated in the pooled action data structure. The data processing system can extract, for each user, relevant information from the received response, and transmit the extracted information to the client device of that user. The data processing system can generate an audio signal corresponding to the extracted information and transmit the generated audio signal to the client device of the user.

By consolidating multiple action data structures into a pooled data structure, the data processing system can allow for accommodating additional service features (e.g., ride sharing or use of coupons or discounts otherwise available only for a relatively large number of transactions). Also, the consolidation of action data structures allows for efficient use of network resources (e.g., reduced number of transmissions of action data structures) and reduction in computational power consumption.

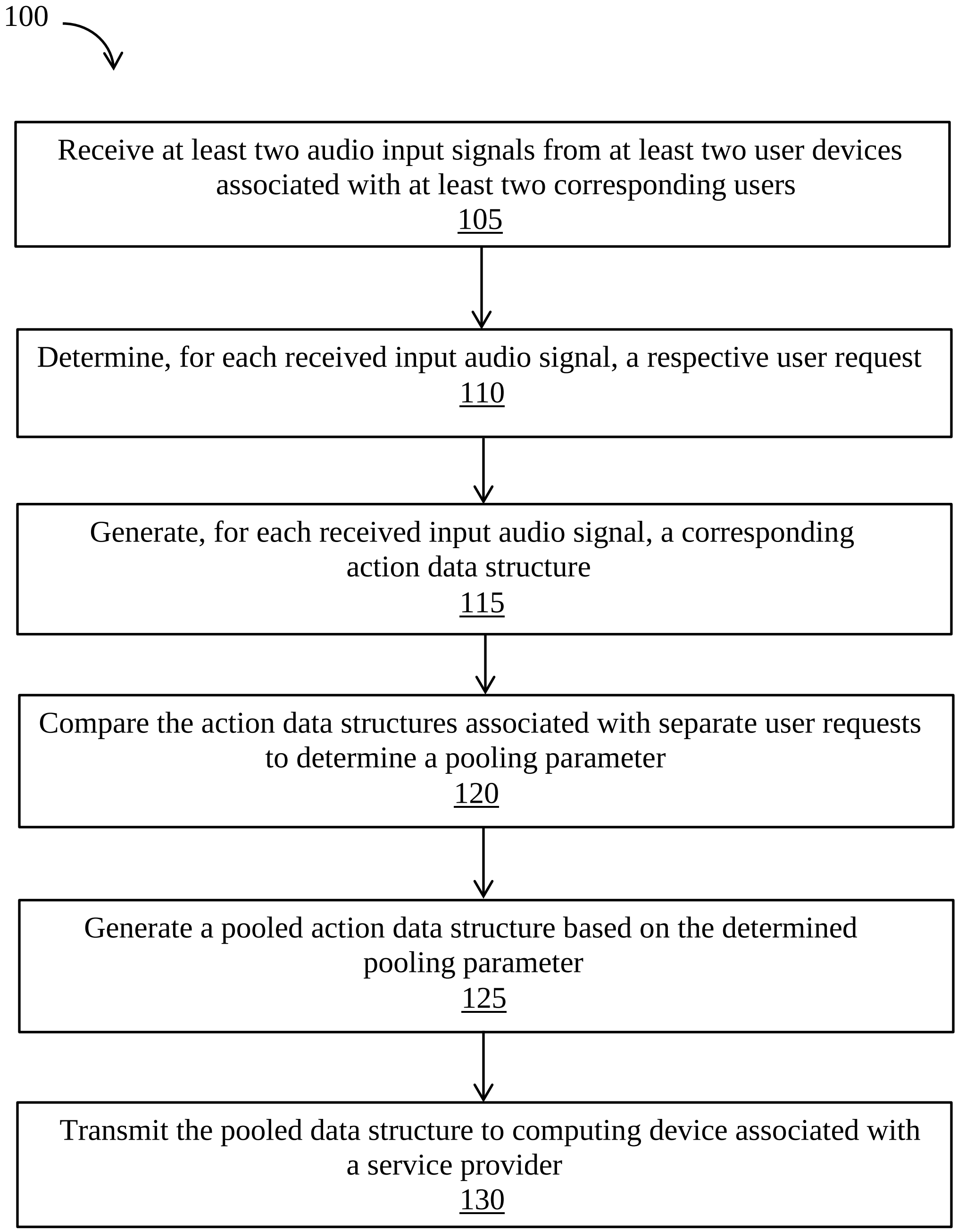
100
Receive at least two audio input signals from at least two user devices associated with at least two corresponding users
105
Determine, for each received input audio signal, a respective user request
110
Generate, for each received input audio signal, a corresponding action data structure
115
Compare the action data structures associated with separate user requests to determine a pooling parameter
120
Generate a pooled action data structure based on the determined pooling parameter
125
Transmit the pooled data structure to computing device associated with a service provider
130

FIG. 1