Technical Disclosure Commons

Defensive Publications Series

April 09, 2015

PROXIMITY BASED SOCIAL SEARCH

Justin Lewis

Gavin James

PROXIMITY BASED SOCIAL SEARCH

ABSTRACT

The invention discloses a proximity search system. The system receives a search query from a first user to search for a second user. The system then identifies one or more candidate users in proximity to the first user. Further, the system transmits a proximity token to the respective client devices of the one or more candidate users. The system further causes the client devices to broadcast the proximity token. The system then determines that the first user's client device detects the proximity token broadcasted by a second user's client device. The system provides the first user with search results that include the second user.

PROBLEM STATEMENT

With increasing usage of social networks, users try to expand their social graph by adding new friends, connections, or followers. In order to add new connections, users run a social search query including a user identifier, e.g., username or real name, of a desired connection. Traditional implementations of social search engines accounts for information in the user's social graph. The information in the user's social graph may be used to determine relevance of results to be returned in response to the search query. Hence, the social search engine may list the connections present in the user's social graph. However, if the user searches for a connection who is not in the user's social graph, the user might face difficulty in finding the desired person in the social network. There is also no mechanism for sorting search results based on proximity of the candidate connections to the user running the search query. Accordingly, a method and system that simplifies how users add connections in a nearby proximity is described.

PROXIMITY SEARCH SYSTEM

The system and techniques described in this disclosure relate to a proximity search system. The proximity search system can be implemented for use by a client device connected to one or more other devices via the Internet, an intranet, or in any client-server environment. The system can include program instructions implemented locally on a client device or implemented across a client device and server environment. The client device can be any electronic device such as a mobile device, a laptop, a smartphone, a tablet, a handheld electronic device, a computer, or a wearable device, etc.

Fig. 1 illustrates an example method 100 for providing proximity based search results to a user searching for another user. Method 100 can be performed by the proximity search system.

The system receives a search query from a first user to search for a second user (110). The system may receive the search query from the first user via a search platform, e.g., search website/application, social network, or any other platform capable of returning users based on the search query. The search query may include a user identifier, e.g., username, public name, vanity name, social name, or any other identifier associated with the second user. The search query can even be a partial user identifier of the second user.

The system then identifies one or more candidate users in proximity to the first user, in response to the received search query (120). On receiving the search query, the system may compare the received user identifier in the search query with known identifiers for a set of users to identify users with the same or similar identifier. For example, if the first user searches for the user identifier “Sam,” the system determines similar identifiers for other users, e.g., Sami,

Sammy, Sambro, Sam Chi, Sam Smith, Sam Haris, or Micheal B. Sam. From the other users with similar identifiers to the queried identifier, the system identifies one or more candidate users that are in proximity to the first user. In an example, the system determines candidate users within a predetermined distance from the first user. The system can automatically set the predetermined distance or the user can input the predetermined distance into the system via a settings menu. For example, the user can input the predetermined distance as 3 miles from the current location of the first user. The predetermined distance can be stored in a memory accessible to the search engine, the first user's client device, in a cloud server, or in an account associated with the first user, etc.

The system can determine a location of the first user and the candidate users based on geolocation information received from various sources, for example, global positioning system (GPS) coordinates from one or more electronic devices associated with the first user and the candidate users, locations tagged in their social media posts, and other user data, e.g., based on receipts or travel information as indicated by e-mails or e-tickets and search queries that identify locations, etc.

Upon determining the one or more candidate users, the system may transmit a proximity token to the respective client devices of the one or more candidate users (130). The proximity token can be used as an access control that can be passed between nodes, e.g., client devices, in order to authorize the nodes to establish connection with each other. The proximity token can take a variety of different forms, e.g., public key, random number, an audible proximity detection token, a WiFi/GPS/Bluetooth proximity detection token, or cryptographic keys. The system can transmit the proximity token using wireless connections, e.g., WiFi, cellular service, satellite

communication, Bluetooth, Internet, intranet, or any client-server environment. The system may also transmit the proximity token to the first user's client device.

The system then causes the client devices to broadcast the proximity token (140). The respective client devices of the one or more candidate users may broadcast the proximity token, e.g., to indicate an acknowledgement or willingness to establish a social connection with nearby users. In an example scenario, a second user's client device receives the proximity token. To allow detection by a nearby user, the system subsequently causes the second user's client device to broadcast the proximity token so that the other client devices in proximity, including the first user's client device, can detect the broadcasted proximity token.

The system then determines that the first user's client device detects the proximity token broadcasted by the second user's client device (150). The system may receive information from the first user's client device that the first user's client device has detected the proximity token broadcasted by the second user's client device. The system then confirms that the first user's client device and the second user's client device share the same proximity token.

On determining that the same proximity token was received between the two devices, the system modifies the relevance of various search results corresponding to the search query from the first user. For example, the system lists the second user's identifier higher in the search results provided to the first user. Hence, the system provides the first user with search results that include the second user (160). Alternatively, the system may choose to provide only the identifiers for the second user in response to the received search query.

Fig. 2 illustrates an example scenario for providing proximity based search results to a user searching for another user as performed by the proximity search system. As illustrated in

Fig. 2, the system receives a search query 250 from user 210 to search for a user A. The system compares the received user identifier in the search query, i.e., user A, with known identifiers for a set of users to identify users with the same or similar identifier. The system then identifies one or more candidate users 220, 230, and 240, from the set of users who are in proximity to the user 210. In an example, the system determines candidate users within a predetermined distance from the user 210. Further, the system transmits a proximity token to the respective client devices of the one or more candidate users 220, 230, and 240. The proximity token can be used as an access control that may be passed between client devices in order to authorize the client devices to establish connection with each other.

The system further causes the client devices to broadcast the proximity token. The client devices may broadcast the proximity token to indicate an acknowledgement or willingness to establish a social connection with nearby users. As illustrated in Fig. 2, the client device of user 220 may be willing to establish a social connection with nearby devices. To allow detection by a nearby user, the system subsequently causes the client device of user 220 to broadcast the proximity token 260 so that the other client devices in proximity, including the client device of user 210, can detect the broadcasted proximity token. On the other hand, the client device of users, e.g., user 230 and user 240, that are not willing to establish a social connection with nearby devices may not broadcast the proximity token. The system then determines that the client device of user 210 detects the proximity token broadcasted by the client device of user 220. The system may modify the relevance of various search results corresponding to the search query from the user 210 based on the proximity tokens. Hence, the system provides the user 210 with search results 270 that include the user 210, e.g., as the top or only returned search result.

Fig. 3 is a block diagram of an exemplary environment that shows components of a system for implementing the techniques described in this disclosure. The environment includes client devices 310, servers 330, and network 340. Network 340 connects client devices 310 to servers 330. Client device 310 is an electronic device. Client device 310 may be capable of requesting and receiving data/communications over network 340. Example client devices 310 are personal computers (e.g., laptops), mobile communication devices, (e.g. smartphones, tablet computing devices), set-top boxes, game-consoles, embedded systems, and other devices 310' that can send and receive data/communications over network 340. Client device 310 may execute an application, such as a web browser 312 or 314 or a native application 316. Web applications 313 and 315 may be displayed via a web browser 312 or 314. Server 330 may be a web server capable of sending, receiving and storing web pages 332. Web page(s) 332 may be stored on or accessible via server 330. Web page(s) 332 may be associated with web application 313 or 315 and accessed using a web browser, e.g., 312. When accessed, webpage(s) 332 may be transmitted and displayed on a client device, e.g., 310 or 310'. Resources 318 and 318' are resources available to the client device 310 and/or applications thereon, or server(s) 330 and/or web pages(s) accessible therefrom, respectively. Resources 318' may be, for example, memory or storage resources; a text, image, video, audio, JavaScript, CSS, or other file or object; or other relevant resources. Network 340 may be any network or combination of networks that can carry data communication.

The subject matter described in this disclosure can be implemented in software and/or hardware (for example, computers, circuits, or processors). The subject matter can be implemented on a single device or across multiple devices (for example, a client device and a

server device). Devices implementing the subject matter can be connected through a wired and/or wireless network. Such devices can receive inputs from a user (for example, from a mouse, keyboard, or touchscreen) and produce an output to a user (for example, through a display). Specific examples disclosed are provided for illustrative purposes and do not limit the scope of the disclosure.

DRAWINGS

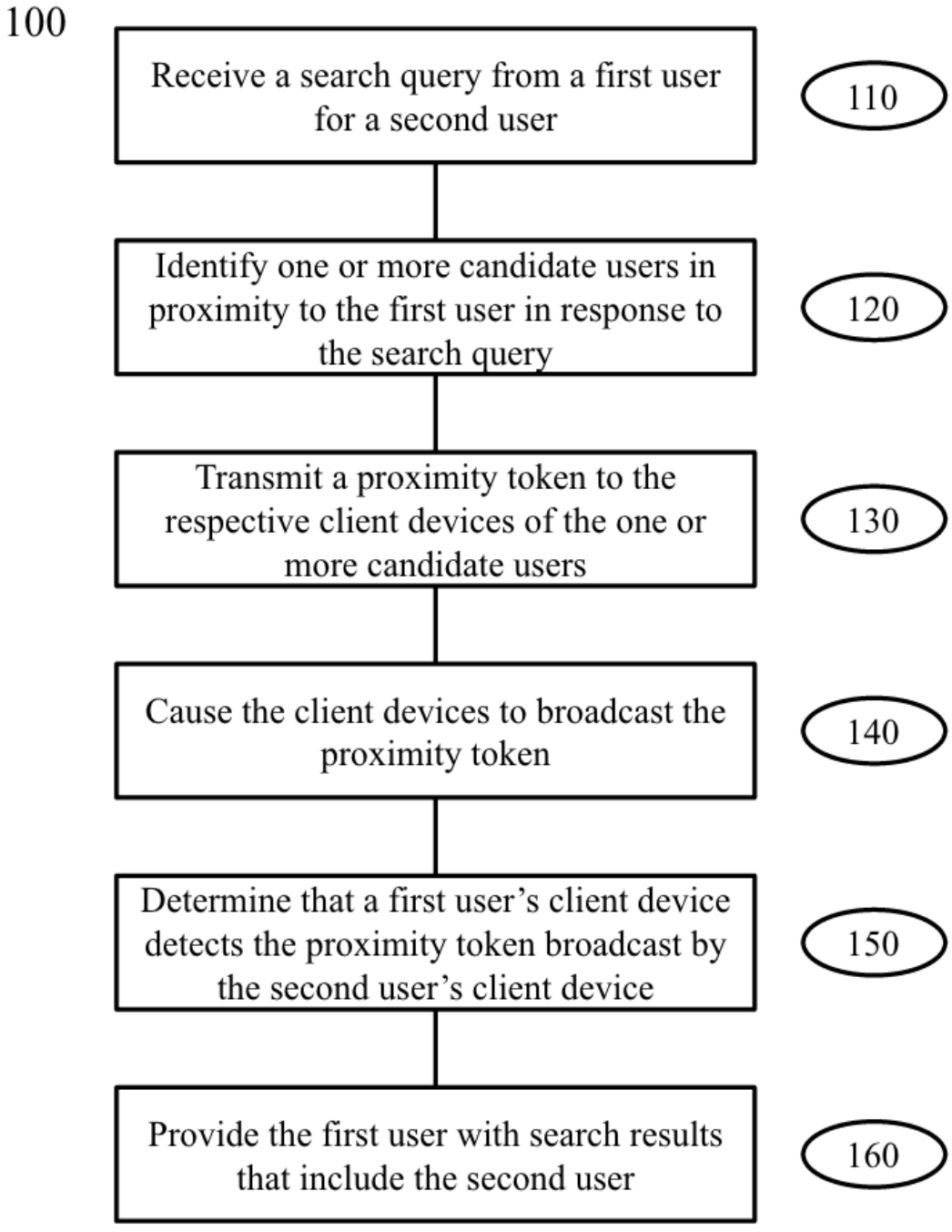

Fig. 1

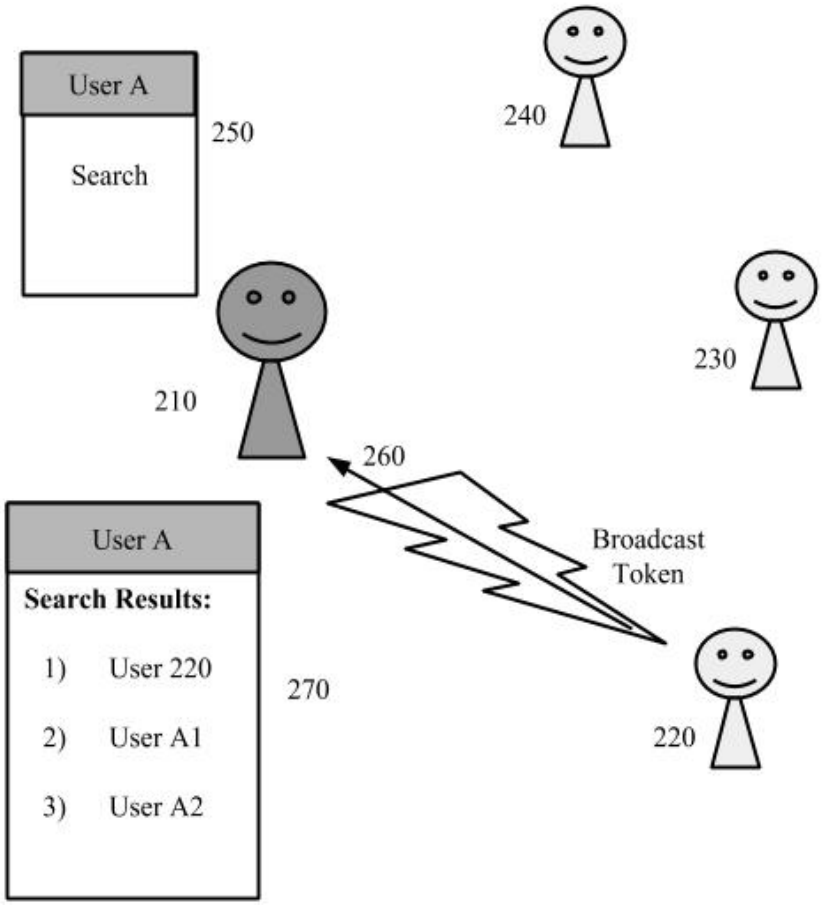
User A
Search
250
240
230
210
260
Broadcast
Token
User A
Search Results:
1) User 220
2) User A1
3) User A2
270
220

Fig. 2

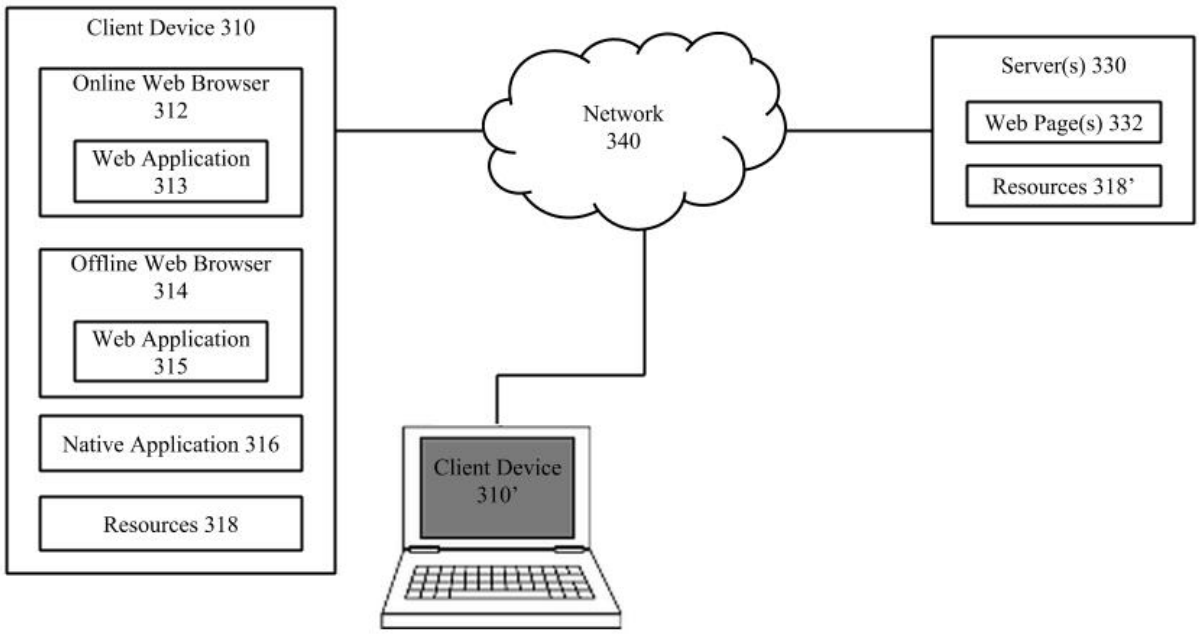
Client Device 310
Online Web Browser 312
Web Application 313
Offline Web Browser 314
Web Application 315
Native Application 316
Resources 318
Network 340
Server(s) 330
Web Page(s) 332
Resources 318'
Client Device 310'

Fig. 3