December 13, 2017

Selective provision of notifications to devices that are offline

Victor Cărbune

Sandro Feuz

Selective provision of notifications to devices that are offline

ABSTRACT

Client mobile devices that are offline are unable to receive notifications via the internet. This disclosure describes techniques to selectively transmit important notifications via alternate channels such as phone calls, SMS, etc. to client mobile devices that are offline. A machine learning model is trained to predict importance scores for incoming notifications, based on training data that includes human-provided ground truth scores. The trained machine learning model ranks the importance of various notifications. The techniques ensure that important notifications are delivered even when a client mobile device is disconnected from the internet.

KEYWORDS

notifications; push API; offline device; notification ranking

BACKGROUND

Many mobile applications ("apps") are designed to operate under the assumption that the mobile device that runs the application is constantly connected to the internet, such that push notifications can be delivered to the app from a server. However, there are some situations, when user mobile devices do not have internet connectivity for longer periods of time. For example, devices may be infrequently connected to the internet when a user travels internationally, e.g., due to high roaming costs. In another example, when the user is in a remote area, mobile internet access can be sporadic. In some instances, users may choose to turn off access to mobile data, e.g., due to high costs. In such situations, apps that are designed with assumption of continuous internet connectivity may provide a suboptimal user experience since relevant information does not reach the user.

DESCRIPTION

This disclosure describes a machine learned model that predicts the importance of information from apps that needs to transmitted to a mobile device of a user, but cannot be sent due to the mobile device being offline. With user permission, the model utilizes various contextual factors to make the prediction. If the predicted importance of the information meets a threshold, one or more alternative channels are utilized to provide the information to the user. The information is delivered using a push-system application programming interface (API).

Fig. 1 illustrates an example environment where entities provide notifications that are to be delivered to a user mobile device to a server that implements the described techniques. The server determines the importance for the various notifications and selectively delivers the notifications to the user's mobile device, e.g., a phone.

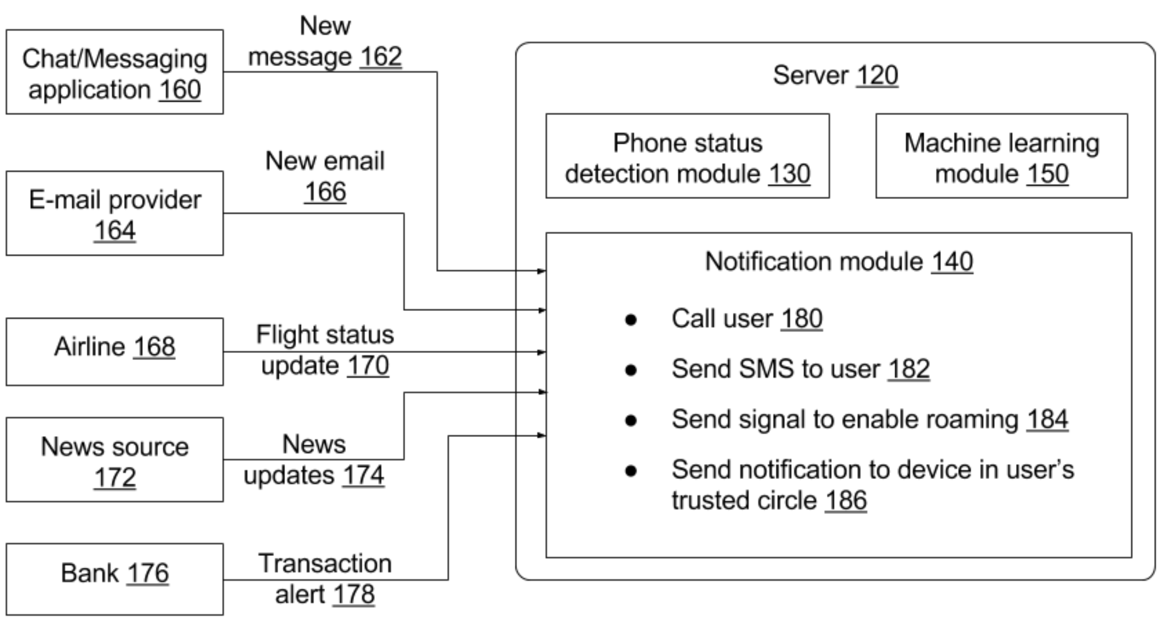

Fig. 1: Machine learning enabled notifications to offline mobile devices

Mobile operating systems provide an API for push notifications. This API is used by various providers, e.g., that provide information to apps on the mobile device, to send notifications via the internet to the apps. As illustrated in Fig. 1, a trained machine learning model is incorporated in a machine learning module (150) in a server (120) that implements the push API. The server (120) also includes a phone status detection module (130) and a notification module (140).

The server receives updates from various providers, e.g., from a chat/messaging application (160), an email provider (164), an airline (168), a news source (172), a bank (176), etc. The updates include push notifications for delivery to a client mobile device, e.g., notifications regarding a new chat message (162), a new email (166), a flight status update (170), a news update (174), a banking transaction alert (178), etc.

The phone status detection module (130) detects the connectivity status of a client mobile device that is associated with the user that is a recipient for the notifications. Whenever the server detects that the client mobile device is unavailable, e.g., for at least a predetermined threshold duration, the client mobile device is determined to be offline. With consent from the user to employ notification techniques described herein, when the client mobile device is offline and one or more notifications are received, the server utilizes the machine learning module to determine an importance for the notifications.

When a notification is determined to be important, the notification module determines a suitable delivery channel based thresholds and user preferences to send the notification to the user. The delivery channels are alternate channels that are available while the client mobile device is disconnected from than the internet. The user can set different thresholds for the different types of escalation, and specify use of the different channels, e.g., based on additional costs for the various channels.

For example, the notification module can deliver the notification via alternate channels as follows:

- **Call the user (180).** A call is placed to the user’s phone number, e.g., for notifications deemed to be extremely important/urgent. The content(s) of the notification or an automatically generated summary of the content is provided to the user via automatically generated speech, e.g., using text-to-speech techniques. When a phone call is used as the escalation path, the spoken interface can additionally provide rich interaction with the user, allowing the user to act on the received high-priority notification. For example, in a scenario where the notification is related to a delayed flight, the spoken interface can prompt the user “Would you like me to connect you with the airline so that you can rebook the flight?” or “Since your flight is delayed, should I inform Fred that you will likely be late for the meeting?”
- **Send an SMS to the user (182).** The content of the notification or an automatically generated summary of the content is sent to the user’s mobile device as one or more short message service (SMS) messages. The SMS can additionally provide rich interaction with the user, allowing the user to act on the received high-priority notification.
- **Send signal to enable roaming (184).** A signal is transmitted (through SMS or a call-like interface) to the client mobile device that instructs the device to enable roaming, which can then enable the server to deliver the notifications to the device using the push API.
- **Send notification to a connected device in user’s trusted circle (186).** With user consent, the server utilizes user contextual information to determine whether a device in the user's trusted circle, e.g. a friend’s phone, is online and in close proximity to the client

mobile device. If a trusted device is identified, the notification is sent to the connected device, e.g., with an indication to relay the notification to the user.

Training of machine learning model

With user consent, the machine learning model is trained using human annotations on different types of data/updates that are delivered to client mobile devices. Different types of information transmitted by the server as notifications to client mobile devices are obtained. The notifications are manually annotated (e.g., by human volunteers) to assign an importance score, e.g., between 0 and 1, where 0 denotes a score for information that is not important and 1 denotes critical/crucial information. The machine learning model is trained using various features of the information as inputs. The features include content of the information. Further, the nature/type of information source (e.g., bank, news provider, airline, etc.) is also provided as an input, e.g., that indicates the general importance of the type of the source. For example, a bank can have a higher bias for important critical notifications than a gaming provider.

When the training data includes additional context factors that are permitted for use in training, such factors can also be utilized to train the model. For example, the factors can include a last known user location, a calendar, etc. Outputs from other machine learned models that are trained to determine use context can also be utilized. User response upon receiving a notification that is in the training data can also be used an input to the model. For example, observing that users take immediate action upon viewing certain types of notification can be an indicator that the particular types are important.

The model is trained to predict the human annotated ground truth importance score of different notifications in the training data. Different model architectures can be used. For example, image based content can be processed using convolutional and pooling layers. For text, a word embedding space can be utilized in combination with a recurrent network such as gated recurrent units (GRU) or long short term memory (LSTM). The output from the machine learned model is a predicted importance score, e.g., between 0 and 1.

Since different providers generate different types of updates in different situations, the importance of the updates may vary in importance, even for updates from a single provider, as described in the examples below.

- An airline transmits notifications, e.g., about the check-in window being open; the flight being ready for boarding; about flight delays and cancellations, etc. In this example, different notifications have different importance. For example, if a flight is being canceled, it may be a high priority notification, since the user benefits from the timely update, e.g., by making adjustments such as booking a different flight, updating calendar availability, etc. With user consent, determination of the importance of the notification particularly can utilize user-permitted contextual factors, e.g., whether the user's schedule includes a meeting immediately after the scheduled flight arrival time.
- A bank typically provides notifications about transactions and account balances; however, the bank can also inform the user about unexpected events such as a credit card or bank account being blocked. The latter type of notification may be more important for the user since it enable the user to take immediate action to prevent the card from being blocked.
- A news provider sends notifications about various news event which may be deemed unimportant. However, news related to emergencies (e.g., a storm warning) in the immediate vicinity of the user, e.g., based on user’s last known location, may be deemed important.

- Chats and emails typically can be delivered when the client mobile device regains internet connectivity. There are certain special cases, however, when it is beneficial for the user to view such messages immediately, even with an additional cost involved. For example, if user provides permission for automatic analysis of incoming messages, messages from trusted senders (e.g., a family member) that include a subject or content that indicates urgency are deemed important. Other examples include receiving an unusually large number of messages from a friend.

The predictions regarding importance of various notifications are also usable to determine how the user receives notifications and content once the client device regains internet connectivity. For example, notifications can be ordered by the predicted importance scores, which can enable the user to navigate effectively through the incoming notifications.

Techniques disclosed in this disclosure are compatible with existing notification filters on a client device. For example, if the user sets a client device to an "alarm only" setting, a priority notification that is escalated and transmitted through alternate channels is subject to an additional ranking, e.g., performed on the client device, to ensure that the user preference to be offline is respected.

When users provide consent, the predicted importance of notifications can be utilized by for various purposes such as network bandwidth conservation, improving content relevance, etc. The disclosed techniques for escalation and the selection of alternate channels can also be implemented as a stand-alone system on its own and be used for other applications.

Further to the descriptions above, a user may be provided with controls allowing the user to make an election as to both if and when systems, programs or features described herein may enable collection of user information (e.g., information about a user's social network, social actions or activities, profession, a user's preferences, or a user's current location), and if 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 information is collected about the user, how that information is used, and what information is provided to the user.

CONCLUSION

This disclosure describes techniques to selectively transmit important notifications via alternate channels such as phone calls, SMS, etc. to client mobile devices that are offline. A machine learning model is trained to predict importance scores for incoming notifications, based on training data that includes human-provided ground truth scores. The trained machine learning model ranks the importance of various notifications. The techniques ensure that important notifications are delivered even when a client mobile device is disconnected from the internet.