September 13, 2018

User-Context Aware Requests for User Feedback

Manikandan Gopalakrishnan

User-context aware requests for user feedback

ABSTRACT

Techniques to improve the process of requesting user feedback for a problem event, e.g., a crash experienced by a user device, are described. The user context is determined and requests for feedback about the event experienced by the user device, e.g., crash reports, are provided at an appropriate time based on the determined context. The request for the user feedback can include a request for a user permission to send captured information about the event, including a device state prior to the crash and a current state of the device.

KEYWORDS

- crash report
- bug report
- device state
- user context
- user feedback

BACKGROUND

Software applications sometimes crash during use, e.g., due to presence of bugs that may survive diligent software testing. When an application crashes, the application vendor or an operating system of the device may request the user to provide feedback that can assist in determining the cause of the crash. For example, such feedback can include device state at the time of the crash, and optionally, user input describing the crash. Further, user permission is sought to send such data to the application vendor.

Currently, such requests for user feedback are provided immediately upon the application crash or error. This may be inappropriate in certain contexts, e.g., when the user is engaged in another activity (e.g., driving, cooking, etc.) and not available to provide user feedback.

DESCRIPTION

This disclosure describes techniques that request user feedback about an application error or crash at a time that is suitable for such a request by taking into account user context. The techniques are implemented with specific user permission to determine user context based on device and determine the action to perform in the event of an error or application crash. In the absence of user permission to determine context, prompts for user feedback may be provided immediately upon an application crash event, and user data is not accessed.

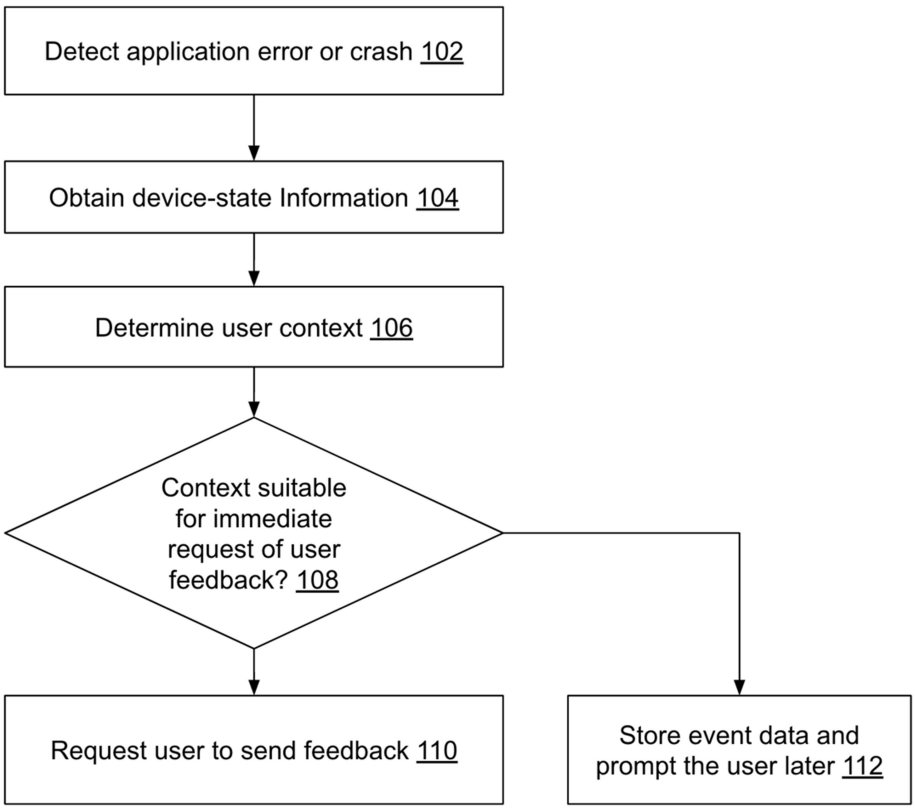

Fig. 1: User-context aware requests for user feedback

Fig. 1 illustrates an example method for providing requests for user feedback in a manner that is aware of the user context. An application error or crash occurs on a user device and is detected by an event handler, e.g., of an operating system of the device (102). The event handler obtains device-state information (104) about the device state that is useful to an application developer to analyze the crash to debug the issue that led to the crash. For example, per techniques of this disclosure, the state of the device just before the crash is determined. Device state is maintained for a sliding time window, and when a crash occurs, the event handler retrieves device state information from the time window prior to the crash.

With user permission, the event handler also determines current user context (106) for the user of the device. For example, an on-device learning system in the device is implemented that utilizes the current device state and historical data (permitted by the user) to determine the user context. Based on the determined context, it is determined whether the context is suitable for immediate provision of user feedback (108). For example, the learning system may utilize the device state, including device location, speed (whether the location is changing, which may be indicative of a driving or in-transit activity), the applications that are currently running, whether the user is making use of the device actively (e.g., via a touchscreen or other physical input device), etc. The learning system can also be fed with developer configurations, e.g., the developer may indicate whether immediate feedback is necessary and optionally, indicate details that are of value in the debugging activity.

If the context is suitable, the user requested to send the feedback (110). For example, the request may include a request for permission to send portions of the device-state information, and optionally, a request for user input regarding the crash. If the context is unsuitable, the event

data is stored, and the user is prompted at a later, suitable time to provide the feedback about the crash (112).

The aforementioned learning system can include, e.g., regression learning models, neural networks, etc. Example types of neural networks that can be used for the classifiers include long short-term memory (LSTM) neural networks, recurrent neural networks, convolutional neural networks, etc. Other learning systems, e.g., hidden Markov models, support vector machines, random forests, boosted decision trees, etc., can also be used.

The described techniques can be implemented as part of an operating system, e.g., of a portable device such as a smartphone, tablet, or laptop computer; a portable speaker; or other consumer device.

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

Techniques to improve the process of requesting user feedback for a problem event, e.g., a crash experienced by a user device, are described. The user context is determined and requests for feedback about the event experienced by the user device, e.g., crash reports, are provided at an appropriate time based on the determined context. The request for the user feedback can include a request for a user permission to send captured information about the event, including a device state prior to the crash and a current state of the device.

REFERENCES

1. Noth, Michael, Jessica Lynn Gray, and Carlos Valdivia. “Collecting application crashes on resource-constrained devices.” U.S. Patent 8,473,251, issued June 25, 2013.

2. Phillips, Derek J., and Robert J. Kroeger. “Tracking remote browser crashes via cookies.” U.S. Patent 8,745,202, issued June 3, 2014.