November 22, 2017

# Dynamic context-based app permissions

Sandro Feuz

Victor Carbune



# Dynamic context-based app permissions

ABSTRACT

Mobile applications obtain permissions from users to access device sensors and APIs. Once a user provides or denies permissions, the choice is permanent, unless the user chooses to change it manually and use of sensor data by the app is not within the user's control. The techniques of this disclosure enable dynamic management of permissions provided to application software, e.g., mobile apps, based on user context. With user permission, a machine-learning model is trained based on contextual data and events, and corresponding app permission settings. The trained model provides predictions of when modifications to app permissions may be suitable. Based on the model predictions, users are notified to make such changes. If users permit, such permission changes are performed automatically when the prediction is associated with high confidence.

KEYWORDS

- App permissions
- Sensor access
- User context
- Mobile apps
- Privacy

## BACKGROUND

Mobile applications obtain permissions from users to access device sensors and APIs. For example, such permissions are obtained when a mobile application ("app") is installed or launched for the first time. Once a user provides or denies permissions, the choice is permanent,

unless the user chooses to change it manually. Regardless, once an app has been provided such access, use of sensor data by the app is not within the user's control.

Therefore, at the time of providing app access to sensor data to an app, users need to evaluate the trade-off between user concerns, e.g., privacy, regarding such data and trustworthiness and functionality of the requesting app, e.g., which may be enabled or enhanced with access to sensor data. The sensitivity of data produced by a sensor is typically context dependent. For instance, a user may consider location data sensitive when the user is at or near a location that is considered private. Such locations are user specific. In another example, a user may allow an app to use data from a device microphone in most contexts, but may desire to deny such access at certain times, e.g., during a meeting, or at certain locations, e.g., a workplace, etc. In another example, a user may provide a spam-detector apps access to incoming phone calls, e.g., thereby benefitting from the detection of spam or unwanted calls. However, the user may sometimes need to temporarily restrict such access, e.g., when conducting a call that is private and confidential. Current mobile operating systems do not have features that enable control of app permissions based on context.

## DESCRIPTION

Users benefit from being able to control permissions granted to apps based on context. Fig. 1 illustrates a mobile device (100) that implements context-based app permission setting per techniques of this disclosure. A permission predictor (104), e.g., implemented as part of a mobile operating system, employs a trained machine-learning model (106) to enable dynamic control of permissions granted to applications (102). The permissions may be, for example, for an application to access mobile device sensors (110A-110D), e.g., location sensors (110A),

movement sensors (110B), camera (110C), etc., and other sensors (110D) device data sources and/or application programming interfaces (APIs).

On-device training for machine-learning model is performed based on user context data (108), when user permit use of such data for training the model. For example, such data can include prior user actions to set or change permissions for apps. The permission predictor can be implemented as an add-on layer to permissions settings that are typically managed by an operating system. The machine-learning model is trained to provide predictions of when the permission settings for an app are to be changed.

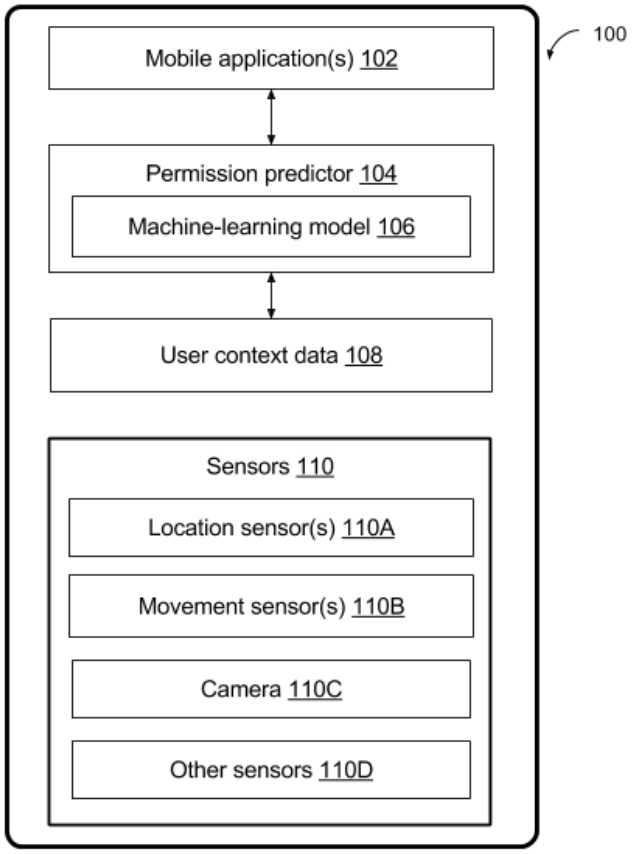


**Fig. 1: Mobile device with permission prediction**

With user permission, the on-device machine learning model is trained on data generated by users during their use of the mobile device. For example, events during which users change

permissions settings for an app are recorded. Examples of such events or user actions include a user revoking permissions previously granted to an app, the user granting specific permissions to an app, etc. Other events, e.g., the user enabling/disabling a sensor temporarily, are also used for training. Some events may be indicators of user intent to deny an app access to a sensor or data source temporarily, e.g., users quitting a long-running app, clearing app data, uninstalling or reinstalling an app, etc. and are also used to train the model.

With user permission, user context data, i.e., events, relevant permissions, and apps that are impacted by events, are provided as features to train the machine learning model. Additional contextual data associated with events that can be used when users provide permissions includes device screen content, running apps, current values of sensor data, e.g., location sensor or accelerometer, features deduced from user context (e.g., calendar entries, conversations, etc.), current time, etc. Some features may be inferred based on such data, e.g., weekday, holiday, etc.

The machine-learning model can be, for example, a feature-based neural network or a recurrent neural network. A recurrent neural network can use the sequential nature of feature data, e.g., points of time when the data was collected, as a time-series for the sequence.

In some implementations, user devices can be provided with pre-trained models (initial models). Such models can be trained on anonymized training data (e.g., obtained from users that provide permission for such use of data) and clustered by a user-embedding vector. The pre-trained models are adapted based on user actions.

The model is evaluated and trained on-device based on current feature data. An important quality of the trained model is precision of the prediction, e.g., if a user confirms a permission change recommended by the model, the particular prediction is considered accurate. When the user rejects the model prediction, such rejection is used as negative examples for further training

of the model. In some implementations, negative examples may be accorded higher weightage than other samples, e.g., if subsequent user data confirms the negative examples. While the foregoing discussion describes use of a machine-learning model, heuristic techniques can also be used for prediction of app permission settings.

When the model determines that permissions previously provided to an app should be revoked, e.g., temporarily in the current context, the mobile device provides non-intrusive notifications to the users to modify the app permissions. Alternatively, if the user permits, the permissions can be revoked automatically, e.g., when the model prediction meets a threshold.

This disclosure provides two key benefits over the current manual permission settings. First, users can more easily and quickly enable or disable permissions for apps based on the model's contextual recommendations or configure the device to automatically manage permissions. Further, users are protected when they forget to temporarily revoke a permission. In the second instance, a user that previously temporarily revoked the permission in similar contexts receives a suggestion from the trained model for such revocation.

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

The techniques of this disclosure enable dynamic management of permissions provided to application software, e.g., mobile apps, based on user context. With user permission, a machine-learning model is trained based on contextual data and events, and corresponding app permission settings. The trained model provides predictions of when modifications to app permissions may be suitable. Based on the model predictions, users are notified to make such changes. If users permit, such permission changes are performed automatically when the prediction is associated with high confidence.