June 2022

Predictive Activation of Device Components Based on Partially Decoded Command

D Shin

Predictive Activation of Device Components Based on Partially Decoded Command

ABSTRACT

Users often invoke a virtual assistant to quickly accomplish routine tasks that require the corresponding components of a device to be woken up from an idle state prior to performing the desired action. Activating the components introduces an inherent latency that can noticeably raise the task completion time and degrade the user experience. This disclosure describes techniques to reduce the latency involved by proactively waking up idle device components by predicting the intended task based on the initial words in a user's spoken command to a voice based virtual assistant. The net effect of starting the wake-up process based on partial decoding of the spoken query can reduce latency without affecting power consumption.

KEYWORDS

- Spoken command
- Voice command
- Command recognition
- Command decoding
- Partial decoding
- Predictive wake
- Virtual assistant
- Smart speaker

BACKGROUND

Users often invoke a virtual assistant to quickly accomplish routine tasks, such as taking a photo, playing a song, making calendar entries, etc. Many of these routine tasks require the corresponding component(s) of a device to be woken up from an idle state prior to performing the desired action. For example, taking a photo via a smartphone requires waking up the image signal processor stack from its idle state in order to capture a photo.

Typically, users perform specific tasks associated with some device components relatively infrequently. For example, an average user may take photos only a few times a day.

Components of the device that are not in active use are put in idle state to conserve power and prolong battery life. Waking up a component from idle state prior to performing the corresponding task has an inherent latency that can sometimes be as high as one second. The latency can noticeably raise the completion time of the task from initiation to completion, thus degrading the end-to-end user experience (UX).

User commands to a virtual assistant are typically processed via a serial pipeline in which the full command is processed and decoded prior to initiating any further action. For instance, if the user instructs a voice assistant to “take a picture,” the entire sentence is decoded in full and then the photography-related component(s) of the device are woken up to fulfill the command.

DESCRIPTION

This disclosure describes techniques to reduce the latency involved in waking up idle device component(s) required to carry out a task that a user requests via a voice-based virtual assistant. To that end, the speech of the user command is decoded in a word-by-word manner rather than awaiting completion of the full spoken command prior to decoding it. Each decoded word is used to predict the likely user intent. Any idle device components or subsystems that correspond to tasks that are highly likely are woken up as soon as the relevant word is decoded. The effect of starting the wake-up process prior to complete decoding of the full query is to reduce latency with minimal impact on power consumption.

For example, when a user asks a virtual assistant to “take a picture,” photography-related device components can be woken up after decoding the word “take” because it has a high probability of being followed by the words “a picture” or “a photo.” Beginning the wake up process right after decoding the first word in the user command allows the photography-related components to be ready before the user finishes speaking the rest of the words in the command.

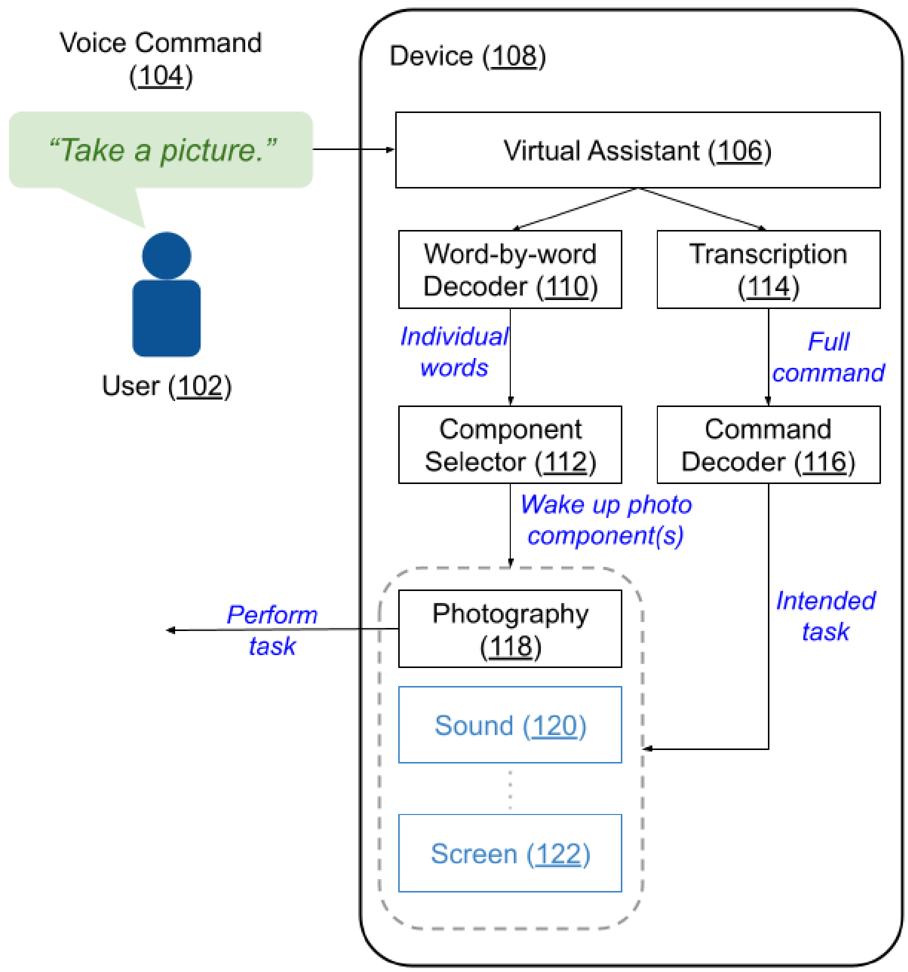

Fig. 1: Early wake up of idle device components by word-by-word command processing

Fig. 1 shows an operational implementation of the techniques described in this disclosure. A user (102) issues a voice command (104) to take a picture to a virtual assistant (106) provided via a device (108). The command is decoded word-by-word (110) with each word passed to a component selector (112) immediately upon being decoded. Based on commands associated with specific words with a sufficiently high probability, the corresponding component(s) can be woken up if they are idle. As shown in Fig. 1, since the word "take" is highly likely to be followed by the words "a picture," components related to photography (118) are woken up. Once the speech of the full command is transcribed (114), it is processed by a command decoder (116) to decipher the user's intended task. Upon confirming that the intended task matches the component(s) woken up, the task is performed as soon as the photography-related components are ready.

In cases where the intended task does not match the predicted task and corresponding device component(s), the incorrectly woken up component is switched back to idle mode and the component(s) matching the intended task are woken up instead. For example, if the user issues the command "take a note," the photography-related components are woken up upon decoding the word "take" because it is highly likely to be associated with the intent to take a photo.

Once the full command is decoded, it is found that the user's intent is to take a note and not a photo. Therefore, the photography related components are switched back to idle mode and the screen components are woken up to display the note-taking application as is the case in normal operation. As a result, false positive predictions have no negative impact on the user experience. As long as the probability of the initially predicted user intent is sufficiently high, the overall benefits of reduced latency in performing the task by early wake up of device component(s) can outweigh the power consumption costs for occasionally waking up incorrect component(s) because of false positives. The threshold values of probability required in the operation of the techniques can be set by the developers and/or determined dynamically at run time.

Word-by-word decoding can be performed by any suitably trained custom word model, such as a model with a transformer-based architecture. The model can be trained on the most popular initial words for common tasks. Employing a custom model can reduce the processing time and improve accuracy for detecting the initial words in a user command. The likely component(s) corresponding to the decoded initial words can be determined using a key-value lookup table that is non-injective such that multiple words (keys) can map to the same task (value). For example, the commands “take a photo” and “shoot a video” both require photography-related component(s) to perform the desired tasks. Therefore, the keywords “take” and “shoot” in the look-up table can both be mapped to the photography component(s).

The techniques can be implemented to support any commands that require idle to active switching of device components related to a variety of functions, such as photography, sound, screen, etc. For instance, commands that start with words such as “listen” or “play” can be associated with a high likelihood of requiring sound-related component(s) while those that start with “turn on” or “show” can be deemed to require use of the device screen. The techniques can support any device that incorporates virtual assistant functionality. Implementation of the techniques can reduce the latency in using a voice-based virtual assistant to perform common tasks, thus enhancing the user experience.

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 voice commands, 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

Users often invoke a virtual assistant to quickly accomplish routine tasks that require the corresponding components of a device to be woken up from an idle state prior to performing the desired action. Activating the components introduces an inherent latency that can noticeably raise the task completion time and degrade the user experience. This disclosure describes techniques to reduce the latency involved by proactively waking up idle device components by predicting the intended task based on the initial words in a user's spoken command to a voice based virtual assistant. The net effect of starting the wake-up process based on partial decoding of the spoken query can reduce latency without affecting power consumption.

REFERENCES

1. Bringert, Bjorn Erik, Peter John Hodgson, Pawel Pietryka, Simon Tickner, Richard Zarek Cohen, Henrique Penha, Luca Zanolin, and Dave Burke. "Systems and Methods for Continual Speech Recognition and Detection in Mobile Computing Devices." U.S. Patent Application 15/045,128, filed February 16, 2016.