January 24, 2018

Automatic management of multiple contexts

JT DiMartile III

Benny Schlesinger

Automatic management of multiple contexts

ABSTRACT

Users interacting with computing devices often multi-task across and navigate between multiple contexts. Users often report distraction and fatigue due to such switching of contexts. For example, it is hard for users to remember an earlier context and resume work at a later time. It has been argued that a user is better off focusing on one context at a time to improve productivity. However, current computers are blind to the different contexts for a user. Applications and programs relating to several current contexts are displayed without taking into account potential for distractions.

Techniques of this disclosure enable automatic grouping of software applications on a user's computer such that only applications that are relevant to a current context are visible. When switching contexts, the corresponding applications for the context are automatically made available. Such user interface enables a user to focus on a current context is strengthened while providing the ability to switch quickly to other contexts as necessary.

KEYWORDS

- Context switching
- Context management
- Multi-tasking
- User interface
- Window management
- Focus UI

BACKGROUND

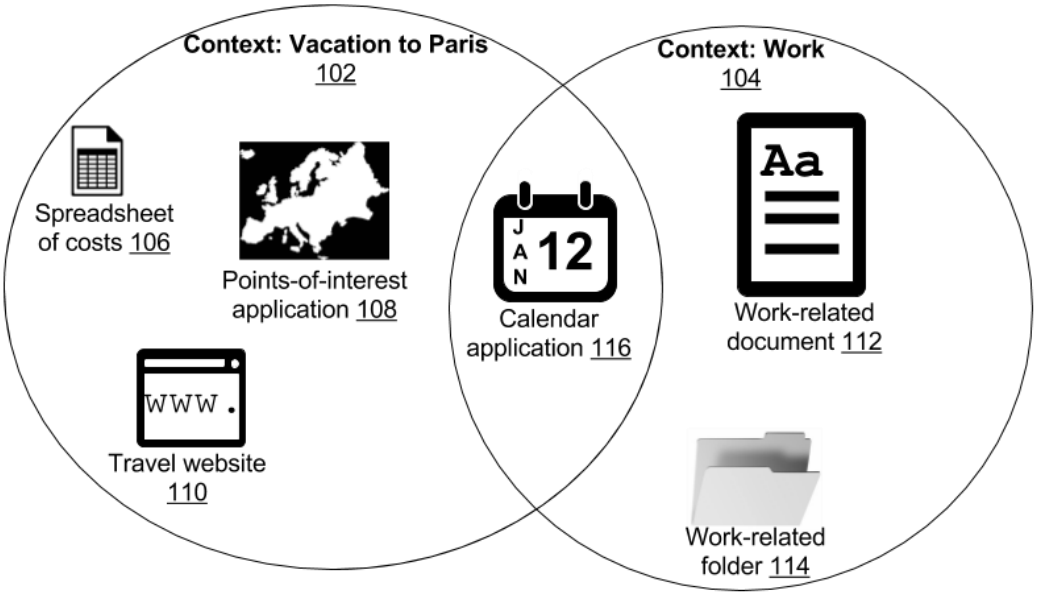

Fig. 1: Multiple contexts on a single screen

Fig. 1 illustrates multiple contexts that share a single workspace or screen. Per the currently open applications, a user is simultaneously planning a vacation and conducting work-related tasks on her device, e.g., personal computer. As illustrated in Fig. 1, the computer displays several applications. For illustration purposes, the applications are shown conceptually grouped into contexts.

A first context (102) comprises applications that relate to the vacation, e.g., a spreadsheet of costs (106), a points-of-interest application (108), a travel website (110), etc. A second context (104) comprises applications that relate to work, e.g., a work-related document (112), a work-related folder (114), etc. There can be applications common to multiple contexts, e.g., a calendar application (116) to coordinate official commitments and dates of travel.

Current user interfaces do not group applications into contexts. Consequently, the user interface presents to the user all open applications other than those that are minimized or hidden by the user. During use, a user may often traverse contexts, e.g., from a work-related document to a travel website and back. When working in a certain context, the user interface may not

display all applications or resources relating to that context, e.g., when the applications have not been launched and conversely, applications not related to that context may be open, potentially causing distraction. When the user switches contexts, it is often difficult for the user to pick back up where she last left off. This can result in a lack of focus, user fatigue, and loss in productivity.

DESCRIPTION

Techniques of this disclosure enable a computer to present a user interface that is aware of context, when users provide permission for determination of user context. The context includes various entities, e.g., open applications, files, documents, browser tabs, media, etc. With user consent and permission, the context is automatically detected using machine learning techniques. The user is enabled to edit context, e.g., by adding or removing entities etc. from the context. The context includes a subset of all open entities. The techniques described herein make it easier for the user to resume work on a certain context by automatically reopening items that were visible from a previous time the user worked in that context.

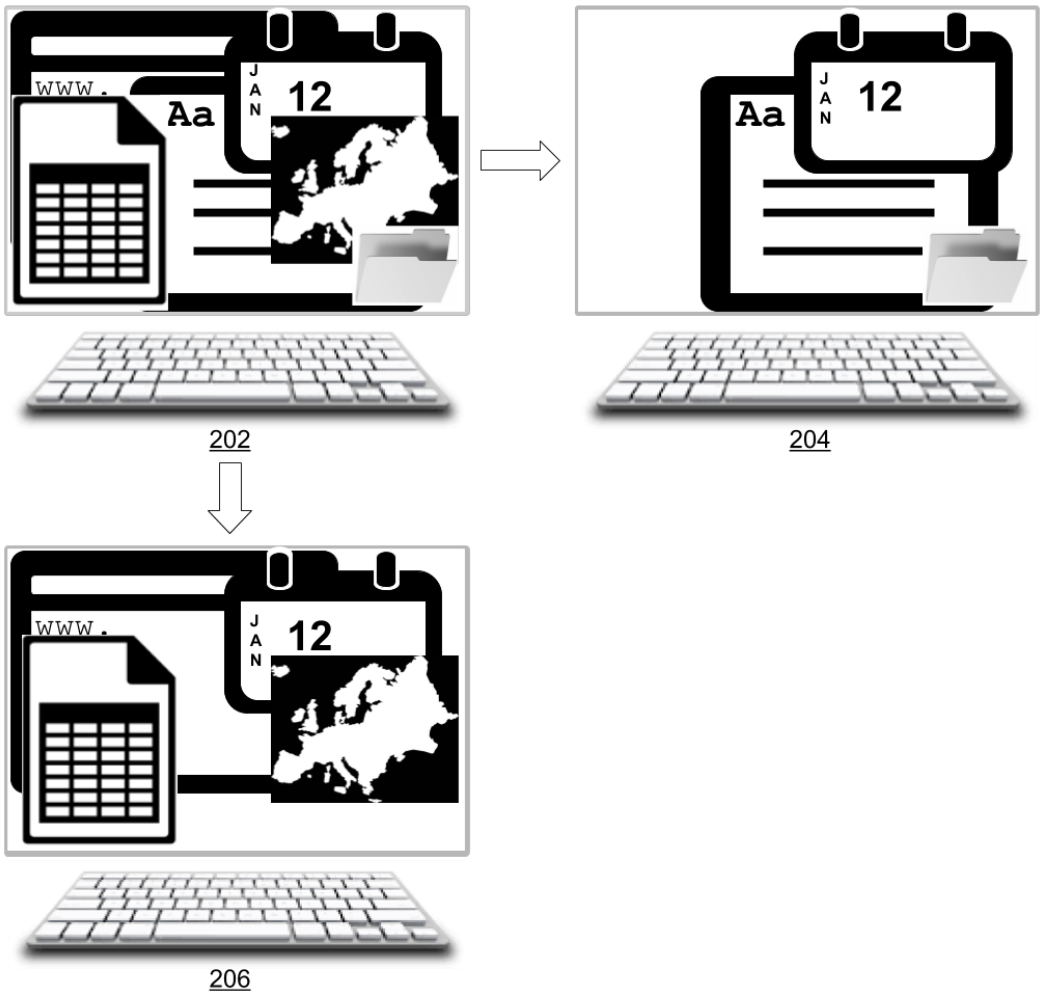

Fig. 2: Automatic detection and enabling of contexts

Fig. 2 illustrates an example of automatic detection and enabling of user interface based on context, per techniques of this disclosure. As seen in the first portion of the image (202), the user interface displays all open applications, including, e.g., work- and vacation-related applications. With user consent, machine learning techniques, the computer groups the open applications by context. Further, the current user activity is utilized to determine whether the user is currently working on work-related tasks, when user provides consent for analysis of user activity.

Upon determination of context, the computer gradually dims and eventually makes invisible vacation-related applications, so that only work-related applications are visible to the user, as seen in a second portion of the image (204). Further, if the computer detects that the user is currently planning a vacation, it makes visible vacation-related applications while dimming and making invisible work-related applications, as seen in a third portion of the image (206). When an application is shared between two contexts (e.g., the calendar application in Fig 1), the application can have a different state in each respective context. For example, the calendar application opens at a different date in each context - the vacation in Paris date in the vacation context and the current week at the office in the work context.

Per techniques of this disclosure, closing a tab or application makes it nominally invisible, but the tab or application retains membership in a particular context. This allows the user to quickly return to the closed application. In this manner, context is effectively a clustering of entities based on a mode of recent access by user, e.g., whether the entities were accessed together at the same time, etc. Context enables a user interface to be presented that enables the user to navigate applications associatively. When a user provides permission, context is stored such that it is retrievable not merely in the current session but in future sessions as well.

Features of the techniques are elaborated upon below.

- *Current context*: The techniques display on the screen, e.g., on the menu bar, the current context determined for the user. When an inference is made that the user has switched contexts, the techniques provide to the user a way to switch to the new context and/or remain in the current context. In case of incorrect switching of contexts, the user can easily navigate back to correct context, e.g., by using the menu bar current-context

indicator. Upon user request, all of the determined contexts can be displayed at once, e.g., equivalent to a context-blind display of applications. When the user switches to a certain context (either explicitly or implicitly), entities that are unrelated to that context is gradually dimmed and eventually hidden. The user can select additional entities to hide anything or bring an application that is currently hidden into the current context.

- *Creation of a new context*: With prior user consent and permission, a new context is created based on applications accessed by the user. The new context can be created either automatically or manually by the user. When a new context is created, the currently open applications are split into two separate contexts. Applications and other entities may span multiple contexts. The system suggests a customizable name for the new context.
- *Reminding users of task completion status*: Using machine intelligence, and with prior user consent and permission, the techniques enable a computer to infer whether a certain task is complete. When a user switches contexts, it is possible that there are incomplete tasks. With user permission, the techniques can detect user fatigue and can provide reminders of incomplete tasks left behind when switching context.
- *Personalization*: When a user provides consent for use of user data to train a machine learning model, such data is utilized to train the model to infer context. The model is trained to determine context and associated computing entities specific to the user. For example, a video of dancing cats may be identified as part of "entertainment" for one user and "work" for another user, e.g., that works in video production of animal shows. Based on user permission, the same video, e.g., dancing cats, is assigned different contexts (e.g., as work or entertainment) for different users.

- *Effect of context on actions*: The current context is provided as an implicit input to user actions. For example, search results are ranked in a manner relevant to the current context, e.g., web searches in the Paris vacation context yield results for businesses in Paris.

By saving context and restoring it when switching, the techniques of this disclosure provide a user interface that can improve user focus and productivity. For example, these benefits accrue as follows:

- *Retrieving historical contextual items*: The techniques enable a user to return to historical contexts, e.g., contexts that existed in past sessions. For example, a vacation-planning user can return to the vacation context days or weeks after initial vacation-related research, and obtain an answer to a specific question such as: "what was that nice hotel I looked up when I started planning my Paris vacation?" This is possible per the techniques of this disclosure because historical context is stored, with user permission, and when retrieved, applications and computing entities relating to that context are presented to the user.
- *Open multiple relevant applications and data at once*: Typically, a user has multiple applications and data while working on a single task. For example, when researching a vacation (a single task), the user may use an internet browser (application) to visit a particular travel site (data) while simultaneously using a spreadsheet program (application) to calculate estimated costs (data). The techniques of this disclosure automatically provide the applications and data for a context when switching into the context. By detecting and grouping applications into contexts, the techniques disclosed herein enable the opening of relevant applications and data.

- *Detecting start of a new context*: Often a user working on a certain task, e.g., a work-related spreadsheet, becomes distracted and opens another application, e.g., a vacation-planning website. The techniques described herein detects the start of a new task, e.g., the opening of the vacation-planning website, and automatically separates the two tasks into different contexts, thereby enabling the user to focus on one task at a time.
- *Reminder of unfinished tasks*: The described techniques treat a distraction (e.g., opening a different application) as a new context. When fatigue is detected, the techniques remind the user of unfinished tasks in other contexts, thereby giving the user an appropriate restart point.
- *Automatic association of tasks to contexts*: When multiple contexts exist simultaneously, the techniques automatically identify the tasks that are related to a particular context.

The present disclosure applies machine learning techniques to group computing entities into contexts, detect the current context of a user, and when appropriate, automatically switch contexts. The machine learning models utilize user data for this purpose, with specific permission from the user. Example features that may be used include the user's current and recent actions, e.g., keyboard or mouse events; computing entities, e.g., desktop, windows, applications, programs, tabs, text, data, etc., that are associated with each other, e.g., that are opened, closed, executed, interacted with or actioned upon in time proximity; input or output events; etc. The machine learning models predict the user's current context, detect a switch in context, whether or not the user completed a task, whether a reminder is to be issued to the user, the appropriate time to issue a reminder, etc.

The machine learning models can be implemented using a long short-term memory (LSTM) neural network. Other types of models, e.g., recurrent neural networks, convolutional neural networks, support vector machines, random forests, boosted decision trees, etc., can also be used. Further, reinforcement learning can be used for training. The described techniques can be implemented as an application (e.g., a browser application), as part of an operating system, etc.

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, user actions and history on a device or applications therein, user's profession, a user's preferences, user's current location, etc.), 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 automatic determination of one or more contexts that each include computing entities, e.g., applications, programs, data, etc. With prior user consent and permission, machine learning models are used to analyze and group the entities computing entities on a computer into contexts, e.g., work, vacation, entertainment, etc. During use, the

current user context is determined, relevant computing entities made available to the user, and computing entities not in the context are dimmed and made invisible. Further, when a switch in contexts is detected, computing entities relevant to the new context are automatically presented and entities associated with the prior context are hidden. The techniques also provide reminders of unfinished tasks. The context-based interaction can improve user focus, reduce distraction, improve task resumption, and can help improve user productivity.