November 10, 2017

On-device text segmentation and user interface with multiple paste options

Matthew Sharifi

Katsiaryna Naliuka

Jan Althaus

On-device text segmentation and user interface with multiple paste options

ABSTRACT

With user permission, on-device text segmentation, e.g., using neural networks, is performed to identify segments within selected text, e.g., contents of a clipboard. The identified segments are provided in a user interface to enable users to perform corresponding actions. For example, multiple paste options are provided, including options to paste entire clipboard contents or an identified text segment. The techniques can be implemented as a feature of an operating system and enable users to take actions with text segments even when software applications do not support partial copying of text.

KEYWORDS

clipboard; text selection; copy paste; text segmentation; text parsing; mobile UI; smart copy

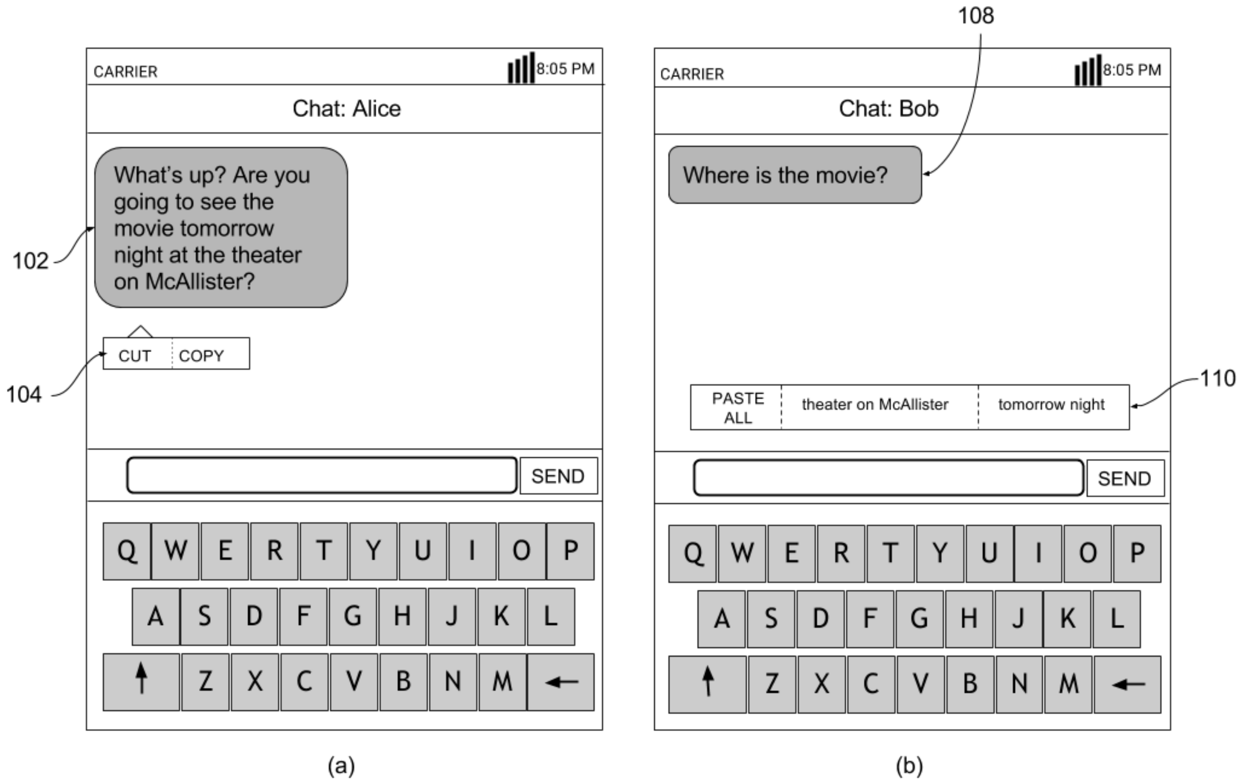
BACKGROUND

Some operating systems include smart text selection features. These system level features enable users to perform smart text selection, such as selecting, cutting/copying, and pasting entities within text. For example, smart text selection makes it easy to select entities such as addresses, phone numbers, etc. within a mobile app and provides actions to send the selections to a relevant app, e.g., a maps application, a phone dialer application, etc.

A problem with smart selection is the lack of consistency across applications. For example, some applications, e.g., mobile messaging apps, override system-level copy and paste. Such apps handle touch events and provide users with options to select and copy a block of text, e.g., an entire message. Thus, such applications only support copy and paste with 1:1 mapping, such that a whole block of text is copied and subsequently available to be pasted. As a result, users are forced to copy the whole message, paste it into an editable text field, and reselect the text that they actually wanted to copy (e.g., an address from a message).

DESCRIPTION

This disclosure describes techniques for parsing and segmenting text entities out of clipboard contents, and a user interface that provides multiple paste options based on the segmentation results.



What's up? Are you going to see the movie tomorrow night at the theater on McAllister?

Where is the movie?

Fig. 1: Copy with multiple paste options

Fig. 1 illustrates an example smart segmentation technique and a user interface that provides multiple paste options. As illustrated in Fig. 1(a), a user has received a message (102) from Alice via a messaging application. The message includes text regarding movie plans, “Are you going to see the movie tomorrow night at the theater on McAllister?”

The user has another chat open with Bob, as shown in Fig. 1(b). Bob has sent a message “Where is the movie?” (108) expressing interest in the movie. Referring again to Fig. 1(a), the user has invoked a text selection menu (104), e.g., to select and copy the message text, to provide a response to Bob. Due to implementation of the copy functionality in the messaging application, the entire message is copied to the clipboard.

Per techniques of this disclosure, when the user provides permission, text selection is analyzed to perform segmentation, e.g., when a user performs a cut or copy operation. The entire contents of the copied text are analyzed, e. g., using an on-device neural network. For example, the on-device neural network is a feed-forward neural network that is run multiple times over the selected text. The neural network is trained to predict text boundaries, and a type for different spans of words. For example, the neural network is trained using available labelled webpage data.

The output from the on-device is analysis is a set of <span, type> pairs. For example, for the text selection of Fig. 1(a), the text selection is the entire message “What’s up? Are you going to see the movie tomorrow night at the theater on McAllister?” In this example, the on-device analysis extracts two entity spans, “tomorrow night,” which has type “date,” and “theater on McAllister,” which has a type “address.”

Based on this analysis, an interface to paste or insert the text is provided when the user initiates a paste action in the chat session with Bob, as shown in Fig. 1(b). Per techniques of this disclosure, paste menu (110) includes multiple paste options. The options include an option to “paste all” that when invoked inserts the entire contents of the clipboard, e.g., message text 102.

The menu includes further options to insert identified text fragments, corresponding to the two entity spans identified using the on-device analysis. For example, the options include “theater of McAllister” and “tomorrow night.”

The segmented paste options can be marked in various ways in the user interface. For example, the options can be marked based on the entity type, based on the first few characters or words from the entity name, etc. Further, the options may be dynamically limited, e.g., based on the type of application or text field where the text is being pasted. For example, if the segmented text includes an entity type date along with other entity types, and it is determined that the paste is being performed in a webpage that includes a form field of type “date,” only the date entity is provided as a paste option.

Further, while Fig. 1(b) shows the segmentation being used in a paste interface, the segmented text can also be used when the user performs a copy operation. For example, when a user performs a copy, possible actions based on part of the clipboard contents are suggested as a popup, notification, or other ways of indicating to a user that a possible action has been detected. The annotation of clipboard contents per the described techniques may be limited to certain applications where it is useful.

The described techniques can be implemented within a software application, e.g., a messaging application, and can also be implemented as an operating system feature. The techniques are implemented upon user permission to perform on-device analysis of clipboard contents and are disabled when the user does not provide such permission.

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

With user permission, on-device text segmentation, e.g., using neural networks, is performed to identify segments within selected text, e.g., within contents of a clipboard. The identified segments are provided in a user interface to enable users to perform corresponding actions. For example, multiple paste options are provided, including options to paste entire clipboard contents or an identified text segment. The techniques can be implemented as a feature of an operating system and enable users to take actions with text segments even when software applications do not support partial copying of text.