

Technical Disclosure Commons

Defensive Publications Series

May 12, 2017

Multiple Simultaneous Responses For Instant Messaging

Nadav Golbandi

Follow this and additional works at: http://www.tdcommons.org/dpubs_series

Recommended Citation

Golbandi, Nadav, "Multiple Simultaneous Responses For Instant Messaging", Technical Disclosure Commons, (May 12, 2017)
http://www.tdcommons.org/dpubs_series/510



This work is licensed under a [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/).

This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.

MULTIPLE SIMULTANEOUS RESPONSES FOR INSTANT MESSAGING

ABSTRACT

A system and method are disclosed for multiple responses in instant messaging conversations. The first part of the system is a machine learning algorithm to predict and identify cases in which a second input box would be needed. The second aspect is a mechanism in the chat or instant messaging application to show an additional text input box. The system includes an algorithm that could collect training datasets from the user's messaging log. From the collected data, a binary classification model could be trained to classify such cases. When the messaging system identifies a similar case it would open a vanish input box so that the user could write his response for the newly received message. Once the response is sent, the new vanish input box would disappear and the user is returned to the original message. The system could be leveraged in any instant messaging solution.

BACKGROUND

In instant messaging conversations, both in one-to-one and group conversations, in which the user is in the midst of typing a response to a message, the user may receive a new message from the other end. Thus, a user might need to make separate responses to either thread, which is a feature that is not enabled by current chat applications. In such situations, the user must either continue with the text that is being typed, which will break the new thread, or respond to the new message. To respond to the new message, the user must delete the already typed text and type in a response. Further, after sending the new message, the user retypes the text that was previously being typed to respond to the first message. Either of these could be tedious or annoying.

DESCRIPTION

A system and method are disclosed that would predict cases in which messaging text that is being typed is required to be stored in order to respond to a new message received after the first one. The first part of the system is a machine learning algorithm to predict and identify cases in which a second input box would be needed. The second aspect is a mechanism in the chat or instant messaging application to show a second text input box.

The algorithm includes collecting training sets based on the user's messaging log. The algorithm assigns values to a target label based on actual user behavior. For cases in which the user has been interrupted with an additional message while in the middle of a response message, the target label is assigned a value of '1'. For all other cases, the target label is assigned the value '0'. The collected data is then used to train any binary classification model. The model could be based on any existing deep learning technologies, for example. The algorithm may identify the need for a second text input box based on the user, context or the features of the conversation. When the algorithm identifies such a case a "vanish input box", a specific UX input box would appear and the cursor is transferred to it. The user may then type a response and send it, while the cursor is directed back to the major input box.

The "vanish input box" as shown in FIG. 1 is provided so that the user could write his response to the newly received message. Once the response is sent, the new vanish input box would disappear and the user could keep working on the original message.

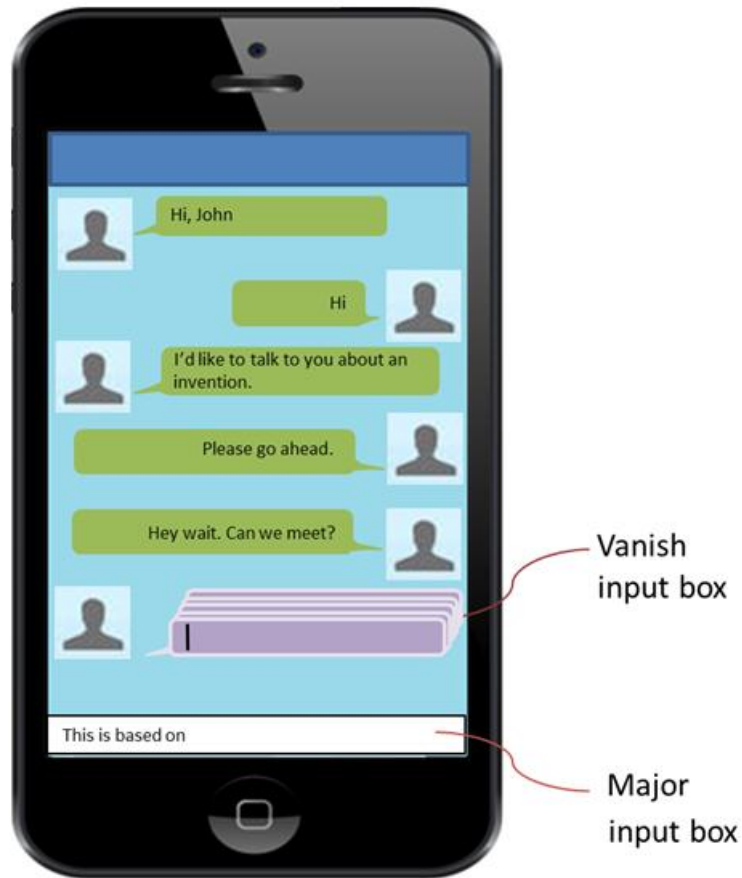


FIG. 1: System of multiple simultaneous input boxes for instant messaging

Further, there may be more than one UX option that could show a flying new input box for the instance response and would vanish once the user had sent the message. The user may also send an empty response to just ignore the new input box. Also, the system includes a multi-editing mode in which the current user message is grayed off, while the cursor is shown in a new line. The user could either type an instance response and send it or just click backspace to get back to the original message that would be active again. The system and method disclosed could be leveraged in any instant messaging solution.