December 2019

Improved Message Facilitator For On-Time And Quick Response

Amy Zhang

Cliff Sze

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

Improved Message Facilitator For On-Time And Quick Response

Abstract:

Increasing reliance on written electronic communication has created a need for automated ways for users to efficiently manage and respond to the various emails, texts, and other electronic messages that they receive. Some solutions have already been introduced to help remind users respond to messages after some number of days have passed, and to help users compose messages. However, through the use of natural language processing, these features can be improved so that users are kept aware of pending response deadlines, and presented with proposed responses that take account of the context of a given message, as well as any questions being asked therein.

Keywords:

Automated, reminder, response, calendar, email, message, messaging, natural language processing, NLP, artificial intelligence, AI, predictive, text, response date, due date, deadline

Description:

As reliance on written electronic communication continues to grow, it is becoming increasingly difficult for users to stay on top of the various emails, texts, and other electronic messages that are received on a daily basis. Some solutions have already been introduced to help in this regard, such as automated notifications that remind a user or "bump up" a message in a user's inbox or message list if the user has not responded to a message after some fixed number of days. Other features have been introduced to aid a user in composing responses to messages

by suggesting various common phrases. While these features are helpful, they can be significantly improved through the use of natural language processing as described below.

Most current systems of issuing message reminders operate based solely on the date the message was received. However, for messages that specify a particular time by which a response is required, these reminder schemes may end up issuing reminders after a required deadline has already passed. Likewise, for messages such as promotions and spam that a user would not normally respond to, these reminder schemes may end up issuing unwanted reminders. By taking advantage of existing classifications (e.g., for spam, promotional messages, etc.), a message system can selectively remind users of those messages that are more likely to be important. Moreover, by employing natural language processing, it is possible for a messaging system to parse the content of messages and identify whether the message requests a response by a certain date. That specific response date can then be used to ensure that a reminder is issued, that the message is "bumped up" in the user's inbox or message, or that the message is otherwise flagged (e.g., by displaying a notation in front of the message such as "due tomorrow"), before the deadline has passed. In addition, as messages with a stated response date are often more critical than others, the message system may also be configured to issue special reminders for such messages. This can include, for example, creating a customized calendar entry, sending an instant message reminder, or adding the reminder to an automated assistant so that the user will not only be reminded of the need to respond to the message, but also realize that the message in question has an actual deadline and is therefore more pressing.

Improvements are also possible in how messaging systems aid users in composing responses to outstanding messages. Some systems may employ natural language processing in this context, but do so only to suggest various suitable generic responses. These generic
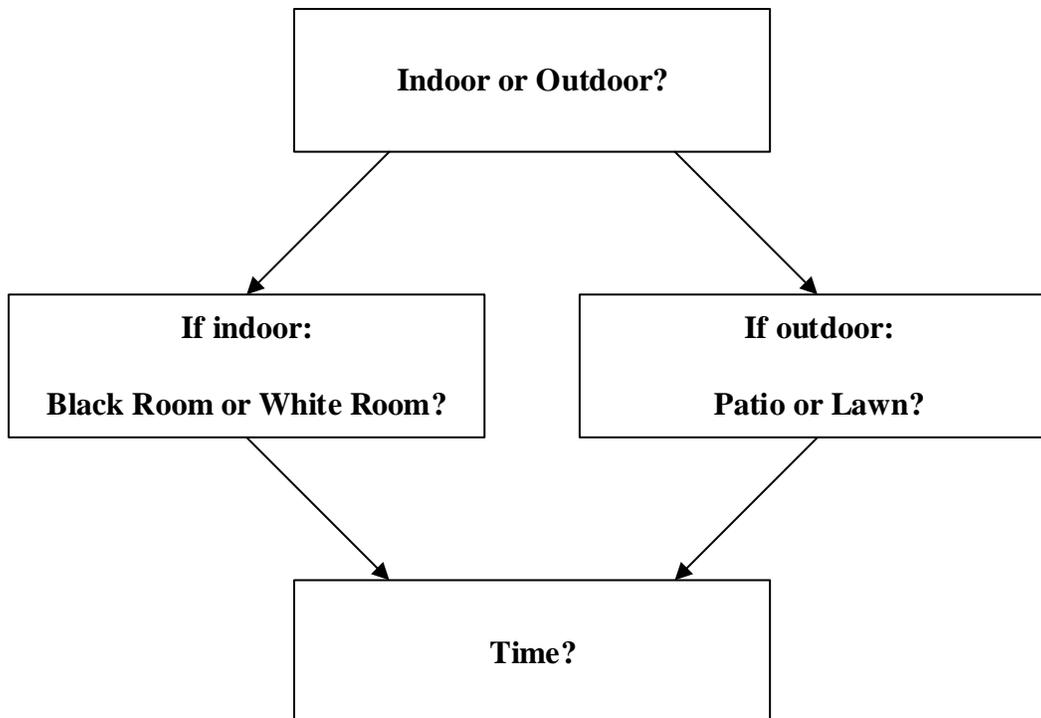
responses can be helpful if the user simply wants to quickly confirm receipt of the message, or

convey their thanks, but are of little use if a message asks for specific substantive information to

be provided.   However, with further development, messaging systems can use natural language

processing to identify questions in a message, and provide a user with an editable response that

only requires certain selections. For example, a user may get the following message:

> Hi Amy,
>
> Thank you for your interest in our venue. Would you like to make a reservation
> for our indoor or outdoor space? If indoor, we have 2 spaces available: the black
> room or the white room. If outdoor, you can choose from either the patio or the
> lawn. What time would you like to make the reservation for?
>
> Yours,
> Cliff

Using natural language processing, the messaging system can parse the text and identify the

following decision tree:

```
                    ┌──────────────────────────┐
                    │    Indoor or Outdoor?    │
                    └──────────────────────────┘
                       ↙                    ↘
  ┌────────────────────────────┐   ┌────────────────────────────┐
  │         If indoor:         │   │         If outdoor:        │
  │                            │   │                            │
  │ Black Room or White Room?  │   │      Patio or Lawn?        │
  └────────────────────────────┘   └────────────────────────────┘
                       ↘                    ↙
                    ┌──────────────────────────┐
                    │          Time?           │
                    └──────────────────────────┘
```

The messaging system can then use the decision tree shown above to solicit answers to these questions. The message system may also utilize other information sources to aid the user in answering the questions, such as by offering a calendar popup so that the user can check her other engagements when selecting an available time, or offering a weather forecast so that the user can decide if an indoor or outdoor arrangement would be preferable. The message system may then provide a template response based on those answers for the user to review and edit, such as the following:

> Hi Cliff,
>
> I would like an indoor space. Please reserve the black room for me on 10/1 from
> 6-9 PM.
>
> Thank you,
> Amy

In some instances, the messaging system may ask the user to answer each of the questions first, and then craft the response based on those answers. In other instances, the message system may present a template response with drop-down menus or clickable options that allows the user to provide the answers. For example, this latter option may be used in conjunction with the reminders discussed above so that a user can open a reminder, review the proposed response, provide any necessary answers, and then hit "send." The message system is able to generate these types of multiple-choice questions using natural language processing, either based on the details in the original email (e.g., indoor or outdoor), or from a database of the most popular answers to a particular question (e.g., most popular soft drinks).

While the above solutions are generally applicable to all messages that a user may wish to reply to, specific considerations arise in the context of messages sent to multiple recipients. In such cases, a sender may ask some questions that apply to everyone, and others that apply to only one recipient. For example, a message may say:

> Hey guys,
>
> I'm going to be in the city from 10/1-10/3. Is everyone available for dinner on 10/1 or 10/2? Let me know by 9/15, as I am going to schedule a business meeting for whatever day we don't meet up.
>
> Cliff, since you're most familiar with the city, do you have a suggestion for where we should eat?
>
> -Amy

In the message above, natural language processing may be used as already described to set reminders for each recipient to respond by 9/15 to let Amy know if they are available, and to compose a template response that identifies whether either or both of 10/1 or 10/2. In addition, natural language processing may be used to further identify that a specific question has been asked of Cliff, and thus to propose a response that includes a blank for his restaurant suggestion. In other instances, a message may be sent to multiple recipients, but a question may be asked of only one. In such cases, natural language processing may be used to identify the one recipient, set reminders, and compose responses only for that one singled-out recipient. This saves the other recipients from being reminded of a message that they have not been asked to respond to. On the other hand, if the singled-out recipient does not respond after some amount of time, the message system may also be configured to remind the other recipients of the message so that hopefully someone in the group will follow-up or prompt the singled-out user to do so.

An exemplary messaging system for performing the methods described herein is shown below in Figure 1. The messaging system may include one or more processors and memory storing instructions and data. In addition, the processor(s) may include a natural language processing module for parsing message text. The messaging system below is shown as a server, which may communicate with various remote client devices such as laptops, mobile phones, and wearable devices over one or more networks. However, various other topologies are also

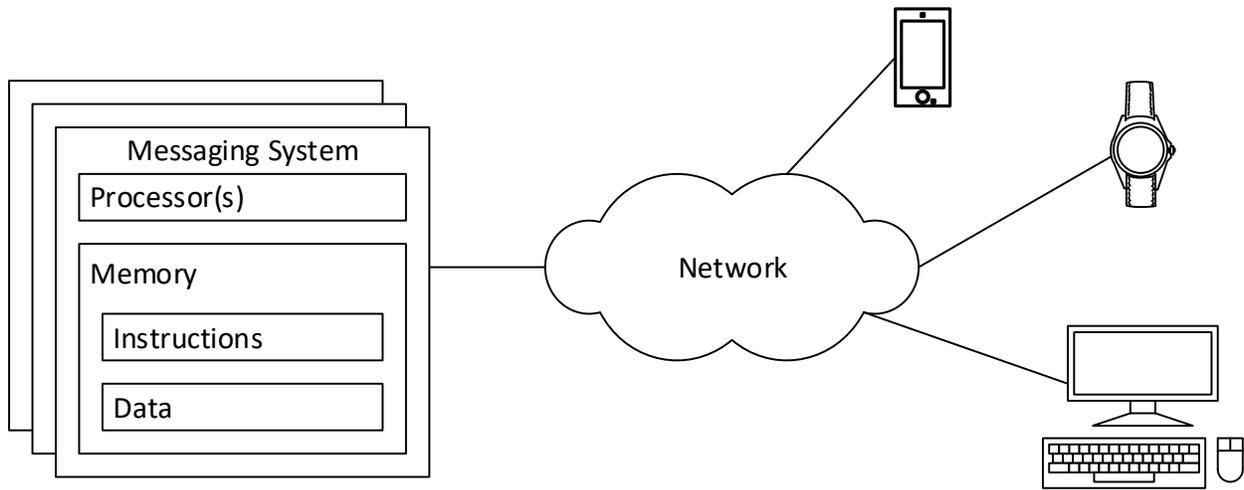possible. For example, the natural language processing and template response creation may instead be performed directly on the client device.

Fig. 1 – Exemplary Messaging System