Responsiveness in Instant Messaging: Predictive Models Supporting Inter-Personal Communication

Daniel Avrahami and Scott E. Hudson
Human Computer Interaction Institute
Carnegie Mellon University, Pittsburgh, PA 15213
{ nx6, scott.hudson }@cs.cmu.edu

ABSTRACT
For the majority of us, inter-personal communication is an essential part of our daily lives. Instant Messaging, or *IM*, has been growing in popularity for personal and work-related communication. The low cost of sending a message, combined with the limited awareness provided by current IM systems result in messages often arriving at inconvenient or disruptive times. In a step towards solving this problem, we created statistical models that successfully predict *responsiveness* to incoming instant messages – simply put: whether the receiver is likely to respond to a message within a certain time period. These models were constructed using a large corpus of real IM interaction collected from 16 participants, including over 90,000 messages. The models we present can predict, with accuracy as high as 90.1%, whether a message sent to begin a new session of communication would get a response within 30 seconds, 1, 2, 5, and 10 minutes. This type of prediction can be used, for example, to drive online-status indicators, or in services aimed at finding potential communicators.

Author Keywords
Statistical models of human activity, Responsiveness, Interruptibility, Availability, Awareness.

ACM Classification Keywords
H5.2. Information interfaces and presentation: User Interfaces; H1.2. Models and Principles: User/Machine Systems.

INTRODUCTION

Inter-personal communication through Instant Messaging, or *IM*, is gaining increasing popularity in the work place and elsewhere. IM programs, or *clients*, facilitate one-on-one communication between a user and their list of contacts, commonly referred to as *buddies*, by allowing them to send and receive short textual messages ("*instant messages*").

Unlike face-to-face communication, users of IM cannot easily detect whether a buddy is available for communication or not. As the use of IM is growing, and in particular in the work place, the inability to detect a buddy's state can often result in communication breakdowns with negative effects on both communication partners. For the receiver, communication at the wrong time might be disruptive to their ongoing work. If, on the other hand, receivers simply decide to ignore communication, the initiator's productivity might suffer as they are left waiting for a piece of information needed for their work.

If, however, we were able to accurately predict whether a user was likely to respond to a message within a certain period of time, then some of these breakdowns could be prevented. For example, models could be used to automatically provide different "traditional" online-status indicators to different buddies depending on predicted responsiveness. Alternatively, models can be used to increase the salience of incoming messages that may deserve immediate attention if responsiveness is predicted to be low. One could also imagine a system whose role is to allow its users to locate others who are available for conversation (for example, to find other users who can provide them with help or support) while hiding those who aren't. This would benefit users looking for help, whose messages would be more likely to get a response, as well as busy users who would be able to stay on task uninterrupted.

The work presented in this paper describes the creation of accurate statistical models that are capable of predicting a user's responsiveness to incoming messages – simply put: whether the receiver is likely to respond to a message within a certain period of time. For example, of the models presented in this paper, one was able to predict with 89.4% accuracy whether a user will reply to a message within 5 minutes and another with 90.1% accuracy a response within 10 minutes (Figure 1).

Background

A number of benefits of using IM have contributed to its increasing popularity. With its near-synchronous nature, IM is positioned somewhere between synchronous communication channels (such as phone or face-to-face) and asynchronous communication channels (such as email, newsgroups, and online forums). This near-synchronous nature allows conversations to range from a rapid exchange

of messages, to hours or even days passing between messages in the same conversation. Since IM is inherently asynchronous, users can choose when or whether to respond to an incoming message. As noted by [25], users welcome the ability to use "plausible deniability" when electing not to respond to messages. IM is thus often regarded as less disruptive than other synchronous communication channels. In fact, IM is sometimes used for communication even between users who share the same physical work-space in an attempt not to disrupt one another's work. This asynchrony means that messages often arrive when a user is engaged in other tasks. Indeed, research shows that users often multitask when using IM [14,20,25]. Particularly in the work place, messages may thus arrive when a user is engaged in important and potentially urgent work.

This means that while it is convenient and desirable for the sender to initiate a conversation, it may be undesirable and often inconvenient for the receiver. The receiver must then choose between staying on task and engaging in conversation. Staying on task and not responding may come at a cost to the initiator, who may need some information from the receiver. The receiver herself may incur a social cost from being portrayed as unresponsive. Engaging in conversation, on the other hand, will often come at a cost to the receiver's ongoing work [27].

One of the most important features of IM clients is the ability to provide some awareness of *presence.* IM clients typically provide this information by indicating whether a user is online and whether the user is currently active or idle (often referred to as the user's "Online Status"). Most IM clients also allow users to set additional indicators to signal whether they are busy or away from the computer. Those, however, are often insufficient as they require users to remember to set and reset them [23]. Begole et al. presented a system that was able to predict a person's presence based on observed patterns [5].

As noted in [4] and [11], knowing whether a person is *present*, however, does not necessarily provide an indication of whether or not that person is *available* for communication. A user who is not present (typically indicated as 'offline' or 'idle') is indeed not available for communication. On the other hand, a user engaged in an important task and unavailable for communication will be indicated by an IM client as present (unless they remembered to manually set their status to 'Busy').

Since the content or topic of an incoming message is typically unknown to the user before it arrives, users generally have to attend to all messages. While the tool presented in [2] increases alerts to some messages based on their content, it does not prevent default alerts from taking place. As a result, users will sometimes elect to turn their IM client off when they are busy, refusing incoming messages altogether [25]. As Isaacs et al note, however, most IM conversations held in the workplace are work-related [20]. This makes closing the IM client a less desirable strategy. Similar to the use of Caller ID in phones, a user can typically also see who the sender of the message is before attending to the message. However, even this brief interruption can, in and of itself, be disruptive [13]. Results from [1] and [8] suggest that, given information about the receiver, senders would be able, and willing, to time their messages to accommodate for the receiver's state.

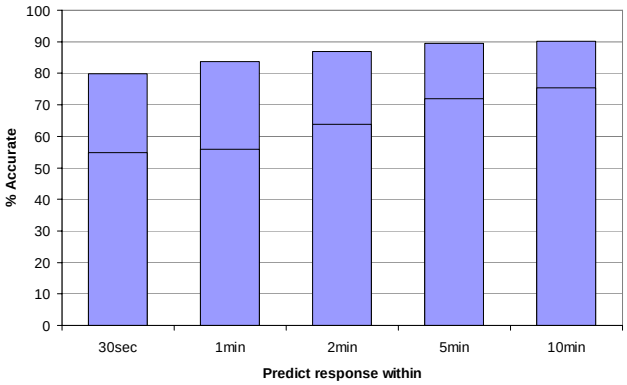

Figure 1. Accuracy of models predicting response to Session Initiation Attempts (SIA-5) within 30 seconds, 1, 2, 5, and 10 minutes. Baseline prior probability is shown with the black lines

Interruptions and Interruptibility

Incoming instant messages join an ever growing number of interruptions a person is exposed to. Those include interruptions external to the computer, such as telephone calls or people stopping by to ask a question, as well as interruptions from various computer applications, including alerts of incoming email, calendar notifications, or notifications of new items from RSS feeds. Unlike face-to-face interaction, most computer-generated or computer-mediated interruptions occur entirely without regard to whether the receiver is ready to accept them.

A number of studies have been performed showing the negative effect of interruptions on people's performance. [13], for example, showed that even a very short interruption can be disruptive, while [7] showed that even an ignored interruption can have a negative effect. Field studies on the effects of interruptions in the workplace observed that, while interruptions can be beneficial to people's work [26], some perceive them to be such a problem that they will physically move away from their computer or even offices to avoid them [18]. In the particular case of IM, we observed a number of managers who refused to use IM for fear of being interrupted.

In previous work [19] we have demonstrated the ability to create statistical models that predicted, with relatively high accuracy, time periods reported by participants as highly non-interruptible. [17], for example, presented statistical models that were able to predict whether a user is "Busy" or "Not Busy" with accuracy as high as 87%.

We wanted to use the knowledge that we gained from the research described above to create useful predictive models in support of inter-personal communication, and in particular IM.

FROM AVAILABILITY TO RESPONSIVENESS

Availability for inter-personal communication is a concept not easy to define. Many factors can contribute to a person's availability: their current mental task, the proximity to the next breakpoint, the identity of the conversation partner, established organizational norms and culture, and so on.

Unfortunately, getting at a person's "true" availability is near impossible. Furthermore, a person's *stated* availability, how available they claim to be, may not match their *demonstrated* availability – their actual responsiveness to communication. For example, a person may be busy and *state* that they are unavailable for communication, while organizational norms coerce that same person to respond to incoming communication, thus *demonstrating* availability. While stated availability is of great interest to us and others, we have decided to focus our initial efforts on predictions of demonstrated availability, more specifically, on the ability to predict *responsiveness* to incoming communication. We are hopeful that this work will allow us to further understand the relationship between responsiveness, demonstrated availability, and finally availability for communication overall.

Behavior as Ground Truth

In order to create a predictive model using machine learning techniques referred to as *supervised learning*, one must first gather data along with *labels* that represent *ground truth* about the data. (Other machine learning techniques, referred to as *unsupervised learning*, that do not use labeled data also exist, but are often less useful for HCI purposes). For example, a set of email messages along with labels provided by a user, indicating messages as either 'spam' or 'legitimate', can be used to train a model to identify spam email messages.

Previous related work, including [9,15,17,19,24], collected naturally occurring behavior as data, using participants' self reports as labels of ground truth. Other work, such as [10] used the behavior of subjects participating in a lab experiment to create their predictive models. The work presented in [9] and [19] (and used by [4] for their models), for example, gathered its labeled data by asking participants, at different intervals, to provide self-reports of their interruptibility on a scale of 1-5. Horvitz et al asked participants to observe video recordings of their day and assign a monetary value to a hypothetical interruption [15], and Nagel et al had participants fill out a short survey on a PDA at random intervals [24].

One of the main drawbacks of using self-reports as measures of ground truth, faced in previous work, is that they are very demanding from the participant's point of view and make it hard to collect large amounts of data. Responding to a voice-prompt (as in [19]) or to a survey on a PDA (as in [24]) or sitting for a long period of time to label past events (as in [15]) can be socially and attentionally costly, and quite time consuming. Another problem with self-reports is that they reflect individuals' subjective interpretation of what is asked of them, an interpretation that can vary from individual to individual.

In contrast with the work mentioned above (and similar to [5] and, for the most part, [16]), the work presented in this paper describes the creation of predictive statistical models trained using *naturally occurring human behavior*. One added benefit of using naturally occurring behavior as the source for learning is that a model deployed as part of a system would be able to continuously observe user behavior to train and improve its performance without requiring any intervention from the user. These considerations led to the design of the data collection mechanism described in the next section.

In the remainder of this paper we describe the data collection method we used and give an overview of the data collected. We then go on to describe in detail the predictive models that we constructed, followed by discussion of the work presented, its limitations, its implications for practice, and conclude with our plans for further research.

DATA COLLECTION

Our data were collected using a background process implemented as a custom plug-in module for Trillian Pro, a commercial IM client developed by Cerulean Studios [6], and running on the Windows operating system. We chose to use Trillian Pro as it supports the development of dedicated plug-ins through a Software Development Kit (SDK) giving access to most of the client's functionality.

Like a number of other IM clients, Trillian allows a user to connect to any of the major IM services (ICQ, AOL, MSN, Yahoo!, and IRC) from within one application. Trillian Pro is further capable of communication with other IM services, including Jabber and Lotus Sametime [22] (used by half of our participants). Using Trillian Pro thus allowed us to recruit participants without concern for the specific IM service they were using. In fact, 8 of the 16 participants used two or more IM services during their participation, and using Trillian Pro allowed us to observe their interactions over all channels.

Another important reason in our decision to use a commercial client such as Trillian Pro, rather than develop a client on our own, was that it provided functionality beyond the simple exchange of text messages. For example, it allows file sharing, audio and video chats, sending images, etc. This reduced the likelihood of participants using other IM clients, which support these features, during the course of their participation in our study.

To capture instant messaging events, as well as desktop events, a copy of Trillian Pro was purchased for each of our

participants and then instrumented with a data recording custom plugin that we wrote. Our plugin is written in C and implemented as a Dynamically-Linked-Library (DLL) that is run from inside Trillian Pro. The plugin automatically starts and stops whenever Trillian Pro is started or stopped by the participant. The following events are recorded:

IM events:

- Message sent or received
- Trillian start or stop
- Message window open or close
- Starting to type a message
- Status changes (online, away, occupied, etc.) of both participants' and buddies'.
- Indicator for incoming message is blinking (if this setting is used)

Desktop events:

- Key press (does NOT include which key was pressed)
- Mouse button click / double-click
- Mouse move
- Window created (including window title and size of window)
- Window minimized (including window title)
- Window in focus (including window title and size of window)
- Window closed

These events, along with the time in which they occurred were saved into log files. These log files were compressed by the plugin "on-the-fly", encrypted, and stored locally on participants' machines.

Participants were required to use Trillian Pro for all their IM interactions for a period of at least four weeks. The compressed log files were collected from participants' computers at the end of their participation and instructions were given to them for removing the plugin.

Privacy of Data

We have taken a number of measures to preserve, as much as possible, the privacy of participants and their buddies. Unless we received specific permission from the participant, the text of messages was not recorded and messages were masked in the following fashion: Each alpha character was substituted with the character 'A' and every digit was substituted with the character 'D'. Punctuation was left intact. For example, the message "This is my secret number: 1234 :-)" was recorded as "AAAA AA AA AAAAAA AAAAAA: DDDD :-)".

Alerts notifying buddies of the participation in the study were sent to each buddy the first time that our participant opened a message window to that buddy and the buddy was online. (A couple of our participants told us that these alerts generated some interesting discussion with their buddies at the beginning of their participation). Buddies of participants who provided the additional permission to record the text of messages were notified with a different alert message that further instructed them of a simple mechanism we included to allow them to temporarily mask messages.

Finally, for determining that two events were associated with the same buddy we used an MD5 cryptographic hash of the buddy name instead of the buddy name itself.

PARTICIPANTS

Data was recorded from 16 participants in two phases. The first phase, which started in May 2005, included eight participants, all Masters students at our department. During their participation, each of these participants was engaged in a number of group projects as part of their studies. Of the participants, six were female and two male, with an average age of 24.5 (SD=2.39, Min=22, Max=29). Six of these participants ran the recording software on their personal laptops. One participant, who used a laptop at school and a desktop computer at home, ran the recording software on both machines. The eighth participant ran the recording software on his account on a shared desktop computer in the Masters students' lab. The remainder of this paper will refer to this group of participants as the "Students" group.

In the second phase, which started in July 2005, we collected data from eight employees of a large industrial research laboratory who used IM in the course of their everyday work. One group consisted of three first-line managers and three full-time researchers. The average age of these six participants was 40.33 (SD=4.97, Min=34, Max=49) with three female and three male. We will refer to these six participants as the "Researchers" group. The second group consisted of two temporary summer interns at the laboratory. Since these last two participants not only worked at the research lab but were also graduate students, we suspected that the patterns of IM use they display will lie somewhere in between that of the Students and that of the Researchers. One female and one male, the last group had an average age of 34.5 (SD=3.54, Min=32, Max=37). We refer to the last two participants as the "Interns" group. All participants in phase 2 ran the recording software on their work laptops. For confidentiality reasons, we did not record the text of messages from any of the participants in the "Researchers" or "Interns" groups.

All of our participants except one were new to Trillian Pro but were able to automatically import the list of all their buddies into Trillian Pro. None of the participants had any difficulty making the transition to using Trillian Pro (and the majority still uses it now after the end of their participation), although some assistance was required with customization of specific options to match the preferences that individual users were accustomed to. All participants ran the recording software for a period of at least 4 weeks. 2 of the participants voluntarily continued their participation for a total of approximately 3 months.

Group	N	Avg age	Total hours in study	Total hours recorded*	Avg hours recorded per participant	Avg Trillian hours per day	Avg active buddies per participant	Total msgs	Avg msg per recorded hour	Minutes per message
Students	8	24.5	9834.5	3839.8	480.0	9.4	31.4	73906	19.2	3.1
Researchers	6	40.3	3709.5	982.5	163.8	6.4	22.8	7290	7.4	8.1
Interns	2	34.5	1593.9	373.0	186.5	5.6	31.5	10343	27.7	2.2
Overall	16	31.7	15138.0	5195.2	324.7	8.2	28.2	91539	17.6	3.4

* - Due to corrupt log files, these numbers are slightly lower than the true value.

Table 1. Overview of the data collected from each group

DATA OVERVIEW

Using Trillian Pro as the client on which we based our data collection resulted in the successful recording of a very high volume of IM events. (A small number of data files were unusable due to corruption in the on-the-fly compression, often as a result of participants' laptops running out of power.) Table 1 provides a summary of data collected in both phases. We collected a total of approximately 5200 hours of recorded data, observing over 90,000 incoming and outgoing instant messages. 73,906 messages from participants of phase 1 spread over 3,839 recorded hours, and 17,633 messages in phase 2 from 1355 hours of recordings. Two of the participants in the Researchers group recorded significantly fewer messages in their logs (96 and 350 messages). However, we did not remove their data from our models and analyses.

To accommodate the fact that data were recorded only when Trillian was running, we provide separate fields in Table 2 indicating the amount of time recorded, as well as the total participation time (calculated for each participant from the start time of their first log file, until the end time of their final log). Since participants in the second phase only recorded activity during business days, their participation time is multiplied by 5/7. The number of recorded hours per day did not vary significantly between groups (p=.23, N.S.).

Participants in the Students and Interns groups exchanged an astonishing average of 19.25 and 19.54 messages per hour recorded respectively. In other words, when Trillian was running, they exchanged, on average, a single message almost every 3 minutes! By comparison, the Researchers exchanged an average of 7.42 messages per hour, or a single message every 8 minutes. Differences between the rates of message-exchanges by group were significant ($F[2,13]=5.08$, $p=.024$). A pair-wise comparison shows that the difference in the rate of messaging was significantly different between the Researchers and either the Students ($t[13]=-2.57$, $p=.023$) or Interns ($t[13]=2.71$, $p=.018$). There was no significant difference between the Interns and the Students groups (p=.32, N.S.).

Overall, message exchanges between our participants and their buddies demonstrated patterns of bursts of rapid exchanges followed by periods of inactivity. Figure 2 shows the delay between 500 consecutive messages between one of our participants and one of their buddies. This pattern is similar to the pattern of email exchanges discussed by Barabási in [3].

In our data set, 92% of messages are responded to within 5 minutes (in fact, 50% of the messages in our data are responded to within 15 seconds). This means that a system that always predicts that a user will respond to any incoming message within 5 minutes will be correct 92% of the time. However, the majority of messages occur as part of a rapid exchange of messages – what we will call an IM session. Once a session has been established, responsiveness is likely to be high and can be explicitly negotiated between parties if needed (for example, one could explicitly declare their responsiveness by sending a message saying that a visitor has entered the room). Consequently, predicting responsiveness to an incoming instant message is interesting primarily for messages that can be defined as initiating a new session, rather than those inside a session proper.

Defining IM Sessions

We define an *IM session* to be a set of instant messages that are exchanged within a certain time delay between one another. Unlike a conversation, a session is not determined by the content of its messages. Indeed, a single conversation may extend over multiple sessions, while a particular session may contain many conversations. The

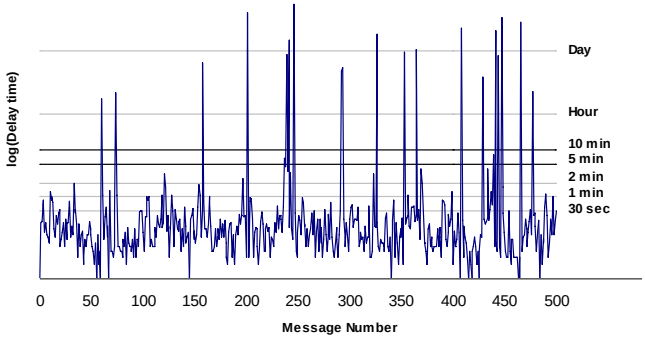

Figure 2. Delay (log sec) between 500 consecutive messages exchanged between one participant and one of their buddies.

main reason for using predictions on sessions rather than conversations in this work is that, even if we had the content of messages from all of our participants, accurately analyzing the content of messages and determining whether two messages belong to the same conversational threads would be quite difficult. We also did not use the closing of a message window to segment sessions since different IM users exhibit different patterns of closing message windows (with some users closing message windows immediately after they send a message, while others keep message windows open for hours with no messages exchanged).

We identify an incoming message from a buddy as a "Session Initiation Attempt" (SIA) if the time that has passed since the participant sent a message to that same buddy is greater than some threshold. In the work presented in this paper we used two thresholds: a 5-minutes threshold (*SIA-5*), similar to the threshold used by Isaacs et al [20], and a more conservative 10-minutes threshold (*SIA-10*). Note that any message identified as a SIA-10 is necessarily also identified as a SIA-5. Of the 45,468 incoming messages in our data, 3,805 were identified as SIA-5 and 3,161 as SIA-10 (both session thresholds are indicated in Figure 2). 72% of messages in SIA-5 and 71% of messages in SIA-10 were responded to within 5 minutes, compared to 92% of the full set of messages. The median response time for messages in SIA-5 and SIA-10 was 37 seconds, compared to the median of 15 seconds for the full data set.

Features and Classes

Before beginning to create the model we processed the raw user-data to produce, for every incoming or outgoing message, a set of 82 features describing IM and desktop states and a set of classes that the models should learn. Table 2a shows a partial list of the IM features associated with every message. We adapted our desktop features from features used in [9] and [17]. Those include the amount of user activity and the most-used application, in the 0.5, 1, 2, 5, and 10 minutes time intervals that precede the message arrival time. We associated applications with a general set of application types (including for example, email, WWW, design-tool, etc.). Table 2b shows a partial list of the desktop features associated with every message.

Day of week
Hour
Is the Message-Window open
Buddy status (e.g., "Away")
Buddy status duration
Time since msg to buddy
Time since msg from another buddy
Any msg with others in last 5 mins
log(time since msg with any buddy)
Is an SIA-5

(a) IM features

App. in focus
App. in focus duration
Previous app. in focus
Previous app. in focus duration
Most used app. in past *m* minutes
Duration for most used app. in past *m* minutes
Number of app. switches in past *m* minutes
Amount of keyboard activity in past *m* minutes
Amount of mouse activity in past *m* minutes
Mouse movement distance in past *m* minutes

(b) Desktop features

Table 2. Partial list of generated features

Our base measure of responsiveness, "Seconds until Response", was computed, for every incoming message from a buddy, by noting the time it took until a message was sent to the same buddy. A histogram of "Seconds until Response" for incoming SIA-5 messages is presented in Figure 3. From this base measure we then created five binary classification labels by indicating, for every message, whether or not it was responded to within each of the following five time periods: 30 seconds, 1, 2, 5, and 10 minutes. (Note that, as indicated in the previous section, less than half the SIA messages were responded to within 30 seconds, while more than half were responded to within the 1, 2, 5, and 10 minutes time periods).

We were now ready to train models to predict each of these binary classifications using the generated features.

MODEL PERFORMANCE

This section presents the performance of statistical models of responsiveness to instant messaging, more specifically to *Session Initiation Attempts* over each of the classes described above. The models presented were generated using a J4.8 Decision-Tree classifier (an implementation of the C4.5 rev. 8 algorithm) using the Weka machine-learning tool-kit [28]. Other classification techniques were also explored but generated models with lower accuracy. For our decision-tree models we used a wrapper-based feature selection technique [21]. This technique selects a subset of the available features by incrementally adding features to the model and testing the model performance until no added feature improves the performance of the model. Each of the models in the process is evaluated using a 10-fold cross-validation technique. That is, each model is created over 10 trials, with each trial using 90% of the data to train, and the remaining 10% to test the model's performance. The overall model accuracy is then presented as the average over these 10 trials. Finally, a boosting process took place using the AdaBoost algorithm [12].

The performance of ten models created for both SIA thresholds and predicting responses within 0.5, 1, 2, 5, and 10 minutes, is presented in Table 3 (labeled "Full Set") and also presented in Figures 1 and 4. The performance is

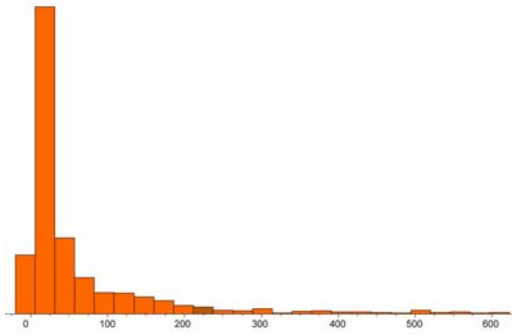

Figure 3. Histogram of "Seconds Until Response" for incoming SIA-5 set with a cut-off at 10 minutes.

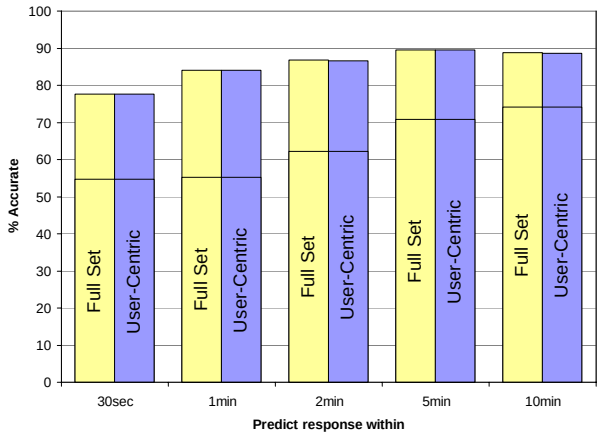

Figure 4. Accuracy (in %) of SIA-10 models compared to baseline by feature sets (Full vs. User-Centric) and prediction class (30secs, 1, 2, 5, and 10 minutes) Baseline prior probability is shown with the black lines

compared to prior probability for each of the predictions. (Prior probability represents the accuracy of a model that picks the most frequent answer at all times). A comparison shows that all models perform significantly better than the prior probability baseline (for SIA-5 models $G^2(1,3805)\geq 1335$, $p<.001$, for SIA-10 models $G^2(1,3161)\geq 916$, $p<.001$). A comparison of accuracy between models created using the SIA-5 and the SIA-10 data sets revealed no significant differences in accuracy.

User-Centric Models

In order to understand the role that buddy state and identity play in our predictions, we next examine ten predictive models of responsiveness created after removing all buddy-related features. We thus term these "user-centric" models.

User-centric models are interesting also as they offer a different solution from a practical standpoint. Models that use the full feature-set (knowing, for example, how much time has passed since the last time a message was exchanged with a specific buddy) may predict, at the same time, different levels of responsiveness to different buddies. In contrast, user-centric models are oblivious to information about the source of the message, and will predict, at any point in time, the same level of responsiveness to all buddies, basing the prediction only on information that is "local" to the user.

A comparison of accuracy between the models presented above and the user-centric models is presented in Table 3. Figure 4 shows a graphical comparison for models created with the SIA-10 set. As expected, the user-centric models performed slightly worse than the models using the full feature set, however this difference was not significant. In fact, in some of the models described earlier, the automated feature-selection process selected no buddy-related features even when they were made available. The user-centric models performed significantly better than the baseline of prior probability in all cases (for SIA-5 models

	Predict response within	30sec	1min	2min	5min	10min
SIA-5	Full Set	79.8	83.8	87.0	89.4	90.1
	User-centric	79.8	83.7	87.0	89.4	89.3
	Baseline	54.7	55.9	63.8	72.0	75.4
SIA-10	Full Set	77.5	84.1	86.7	89.6	88.9
	User-centric	77.5	84.1	86.6	89.6	88.6
	Baseline	54.7	55.1	62.2	70.7	74.2

Table 3. Accuracy (in %) of models compared to baseline by data sets (SIA-5 vs. SIA-10), feature sets (Full vs. User-Centric) and prediction class (30secs, 1, 2, 5, and 10 minutes)

$G^2(1,3805)\geq 1335$, $p<.001$, for SIA-10 models $G^2(1,3161)\geq 916$, $p<.001$). Again, no significant difference in accuracy could be found between SIA-5 models and SIA-10 models.

A Closer Look at Selected Features

Following model generation we examined the features that were automatically selected for the 20 models presented above. These features represent those providing the most useful and predictive information to the model. Models built from the full set of features selected on average 12.3 features, while user-centric models selected, on average, 10.4 features (this difference is not significant).

Most Selected Features

Since the combined total of distinct features selected by all models was high (57 out of the possible 82), for this discussion we group together features describing similar user activity and application information regardless of the time interval they describe (e.g., group all Keyboard Count features together). We further group features into 3 high-level categories: buddy-related IM information, user-centric IM information, and desktop information.

The top 10 selected features for both types of models are:

Full-Data Models	**User-Centric Models**
Mouse Distance Traveled (pix)	Mouse Distance Traveled (pix)
Mouse Event Count	Time Since Last Outgoing Msg
Time Since Last Outgoing Msg	User Input Count
Most Focused Window Type	Most Focused Window Type
User Input Count	Mouse Event Count
Keyboard Count	Duration of Own Status
Time in Most Focused Window	Own Status
Duration of Own Status	Keyboard Count
Time Since Last Incoming Message from Different Buddy	Location (laptop/work/home)
Time Since Last Outgoing Message to Different Buddy	Window Switches Count

Note that the top features selected for both types of models each include six features that are related to desktop activity, (four of which are directly related to user input). This

indicates significant predictive influence from the amount of user interaction. Of features related to IM, the time since the last outgoing message, as well as the duration of the current online-status of the participant appear in both lists. It is possible that the duration of status was frequently selected by our models as it could indicate a recent change of state. Finally, we can see that two features describing IM interaction with other buddies were frequently selected for models built from the full set of features for predictors of responsiveness.

Distribution of Feature Types

Next we examined the distribution of feature selection by high level category. On average, full-set models selected 55.3% desktop features, and 44.7% IM features (22.8% user-centric IM features, and 22% buddy-related IM features). When moving from these models to user-centric models, the distribution of selected features shifts to 62.6% desktop features and 37.4% IM features, suggesting that the void left by the removal of buddy-related IM features was filled, for the most part, by user-centric IM features.

Contribution of Desktop Features by Time Window

As described above, desktop features accounted for over 50% of the features selected by our models. The desktop features we generated looked at different time intervals (e.g., from the last 5 minutes vs. from the last 30 seconds). Figure 5 shows the percentage that features with different time intervals were selected for both full-data models and user-centric models. It is interesting to observe that desktop-features using longer intervals are selected more frequently, potentially because they provide information that is less susceptible to small changes and noise or because longer trends have more predictive importance.

DISCUSSION

In the previous section we have presented statistical models that are able, with high accuracy, to predict responsiveness of IM users. Specifically, these models are able to predict whether a user is likely to respond to an incoming message within a certain time period. Since our participants showed a high level of responsiveness overall, we were particularly interested in predicting responsiveness to messages that represent a buddy's attempt to start a new session (incoming *Session Initiation Attempts*).

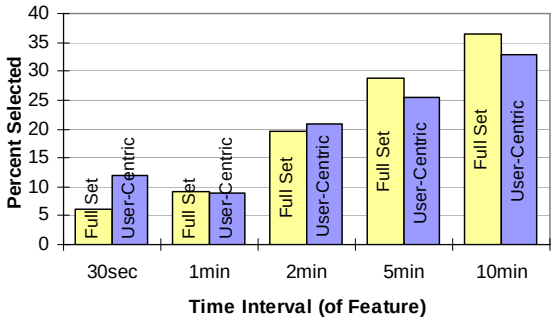

Figure 5. Percent of desktop features selected as a factor of the time interval they were computed on

Indeed, predictive models of responsiveness can be applied in a number of useful ways. For example, models can be used to automatically provide different "traditional" online-status indicators to different buddies. Alternatively, models can be used to increase the salience of incoming messages that may deserve immediate attention (such as in [2]) if responsiveness is predicted to be low. Models could also be used by a system that will show a list of potentially responsive buddies to users who are looking for help or support, while hiding others. We now discuss a number of issues regarding the practical use of predictive models of responsiveness:

Implications for Practice

Preserving Plausible Deniability

One of the key benefits of IM is users' ability to respond to messages at a time that is convenient to them (or even not respond at all). The insufficient awareness provided by most IM clients is at the source of the problem that we are trying to solve with our models. However, it is the ambiguity inherent in this insufficient awareness that provides users with 'plausible deniability'; that is, it allows them to claim that they did not see a message or even that they were not at their computer. It is thus important to warn against a naïve use of predictions of availability. Providing prediction of responsiveness to buddies "as-is", would substantially reduce plausible deniability and should be avoided. Instead, careful consideration of the application and presentation of predictions is required (for an example of the effect of different awareness displays on timing of interruptions see [8]).

Making Predictions Visible to the User

In all current IM clients, users can see their own online-status. This allows them to be aware of and control the presence that they expose to others. Similarly, any system providing automatic predictions of responsiveness to others should reflect this information back to the user. One danger, of course, is that users will attempt to learn which factors determine the system's predictions. For example, in a system that uses responsiveness to determine whether to include a user in a set of possible communicators, a user may try to "game" the system in order to always appear as non-responsive. The system, however, can potentially avoid such a situation by making use of predictions from multiple models. A greater number of models, and potentially a greater number of features, could reduce the overall effect of any one feature in the prediction. Finally, allowing users to override the predictions will likely eliminate the need to "game" the system.

Multiple Concurrent Levels of Responsiveness

In this paper we presented a set of models, which we called User-Centric, generated using only information about the state of the user without any buddy-related features. Our

primary reason was to investigate the relative accuracy of user-centric models. However, the use of user-centric models also has implications for practice. Specifically, a predictive model that takes into account features describing the state and history of a user's interaction with different buddies will, inherently, predict different levels of responsiveness to different buddies. On the other hand models that use only information about the state of the user are guaranteed to provide the same prediction regardless of the identity of the buddy initiating the session. This difference should be carefully considered by the system designer when deciding which type of models to use.

Limitations

One limitation of the models presented is this paper is that they are unaware of the content of messages sent and received. A large number of messages do not in fact require immediate responses. Avrahami and Hudson list different levels of responsiveness expected for different types of messages [2]. A model for predicting responsiveness that does not use the content of messages will use other features to explain the lack of a response, potentially leading to inaccurate predictions.

Predictions of responsiveness without using content may also result in misinterpretations of availability. An example of a case where mere responsiveness incorrectly reflects availability is that of responses used for deferral. For example, a user responding quickly with a message saying "can't talk, in a meeting" would demonstrate high responsiveness but low availability. A model unaware of the content of the message is likely to misinterpret this behavior. In order for such events to be classified correctly they should, more appropriately, be noted in the training data as "no response". This, however, would be impossible to detect without the content of the messages (and even then, detecting those in an automatic way is not trivial).

Future Work

Content Analysis

For future improvements to our models, we plan to look at the content of messages provided by four of our participants. We plan to test the ability to automatically detect the topic of a message. This will allow us to address the limitations discussed above as well as introduce other content-based features to our models.

Responsiveness as a Continuous Measure

Our plans for further exploring the predictions of responsiveness include the creation of models that predict the time until a user responds as a continuous measure. In this paper we presented models capable of successful predictions for 5 different time periods, however, a system might require a model that can provide finer grain predictions of responsiveness. As a first step in this direction we plan to use regression models to try and estimate users' response times. Through the use of linear-regression we hope to also be able to understand the detailed contribution of specific features and the interactions between those features.

Beyond Desktop Events

The work presented, for example, in [4,9,16] described the creation of statistical models that used input from a person's calendar as well as sensors external to the workstation. Those included a door sensor, sensing whether the door was open or closed, a phone sensor, sensing whether the phone was on or off hook, simple motion detectors, and speech sensors, implemented with microphones installed in the person's office, or the microphone built into participants' laptops. When designing the data collection for the work presented in this paper we decided not to use sensors external to the desktop. While we believe that it is reasonable to expect events and activities external to computer usage to be reflected in that usage (for example, a user attending to a visitor is likely to generate fewer computer events), we suspect that improvement to our models could potentially be generated from features that use such sensor data. As the collection of software events is possible on most all computers and is extremely low cost in comparison with other sensors, we plan to investigate the correlation between software generated events and external events.

From Responsiveness to Availability

As we mentioned at the beginning of this paper, we are interested in a better understanding of the concept of availability. In the future we plan to collect both behavioral data (as we did in this work), as well as collect participants' self-reports, in order to understand the relationship between stated and demonstrated availability.

SUMMARY

Instant Messaging is an important communication channel increasing opportunities for inter-personal communication between both distributed and co-located people. The low cost of initiating communication over IM, combined with its currently limited awareness support, results in messages often arriving at times that are inconvenient or distracting for the receiver. An attempt to start a conversation may then either result in a disruption to the receiver's work, or if the receiver decides to ignore it, may result in the initiator left without a needed piece of information. In the work presented in this paper we focused our efforts on predictions of *demonstrated availability* – more specifically, on the ability to predict *responsiveness* to incoming communication. We described the collection of a large corpus of IM interaction and the creation of statistical models that successfully predict a person's responsiveness to incoming messages, in particular responsiveness to incoming attempts at initiating a new IM session. We further investigated the performance differences between models that provide different responsiveness levels for different buddies, versus "user-centric" models that predict the same responsiveness for all buddies. The performance of these "user-centric" models was not significantly

different from that of models that were able to use the full set of features. This means that considerations for the particular use of the models will allow a system designer to choose between these two alternative model types.

Ultimately we are interested in understanding the factors that govern availability (both stated and demonstrated). We believe that the ability to predict the behavioral manifestations of availability, namely responsiveness, advance us in that direction.

ACKNOWLEDGEMENTS

We would like to thank Mike Terry, James Fogarty, Darren Gergle, Laura Dabbish, and Jennifer Lai for their help with this work. We would also like to thank our participants for providing us with this invaluable corpus. Finally we would like to thank the 61C café for providing us with a great environment for writing. This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. NBCHD030010, and by the National Science Foundation under grants IIS 0121560 and IIS 0325351.

REFERENCES

1. Avrahami, D., Gergle, D., Hudson, S.E. and Kiesler, S. Improving the match between callers and receivers: A study on the effect of contextual information on cell phone interruptions. *Behavior & Information Technology* (in press).
2. Avrahami D. & Hudson S. E. QnA: Augmenting an instant messaging client to balance user responsiveness and performance. In *Proc. CSCW'04*, ACM Press (2004), 515-518.
3. Barabási, A.L. The origin of bursts and heavy tails in human dynamics. Nature, 435, Nature Publishing (2005), 207-211.
4. Begole J., Matsakis N. E. & Tang J. C. Lilsys: Sensing unavailability. In *Proc. CSCW'04*, ACM Press (2004).
5. Begole, J., Tang, J.C., Smith, R.E., and Yankelovich, N. Work rhythms: Analyzing visualizations of awareness histories of distributed groups. In *Proc. CSCW'02*, ACM Press (2002).
6. Cerulean Studios - Trillian Pro http://www.trillian.cc
7. Cutrell, E., Czerwinski, M. and Horvitz, E. Notification, Disruption, and Memory: Effects of Messaging Interruptions on Memory and Performance. In *Proc. INTERACT'01*, IOS Press (2001), 263-269.
8. Dabbish L. and Kraut R. Controlling interruptions: Awareness displays and social motivation for coordination. In *Proc. CSCW'04*, ACM Press (2004).
9. Fogarty J., Hudson S. E. & Lai J. Examining the robustness of sensor-based statistical models of human interruptibility. In *Proc. CHI'04*, ACM Press (2004).
10. Fogarty J., Ko A. J., Aung H. H., Golden E., Tang K. P. & Hudson S. E. Examining task engagement in sensor-based statistical models of human interruptibility. In *Proc. CHI'05*, ACM Press (2005), 331-340.
11. Fogarty J., Lai J. & Christensen J. Presence versus availability: The design and evaluation of a context-aware communication client. *IJHCS, 61, 3*, (2004).
12. Freund, Y., and Schapire, R.E. Experiments with a new boosting algorithm. In *Proc. ICML'96*, Morgan Kaufmann (1996), 148-156.
13. Gillie, T., and Broadbent, D. What makes interruptions disruptive? A study of length, similarity and complexity. *Psychological Research, 50, 1* (1989), 243-250.
14. Grinter, R. and Palen, L. Instant Messaging in Teen Live, In *Proc. CSCW'02*, ACM Press (2002), 21-30.
15. Horvitz E. & Apacible J. Learning and reasoning about interruption. In *Proc. ICMI'03*, ACM Press (2003).
16. Horvitz E., Koch P., Kadie C. M. & Jacobs A. Coordinate: Probabilistic forecasting of presence and availability. In *Proc. UAI'02*, Morgan Kaufmann (2002).
17. Horvitz E., Koch P. & Apacible J. BusyBody: Creating and fielding personalized models of the cost of interruption. In *Proc. CSCW'04*, ACM Press (2004).
18. Hudson, J.M., Christensen, J., Kellogg, W.A., and Erickson, T. "I'd be overwhelmed, but it's just one more thing to do": Availability and interruption in research management. In *Proc. CHI'02*, ACM Press (2002), 97-104.
19. Hudson S., Fogarty J., Atkeson C., Avrahami D., Forlizzi J., Kiesler S., Lee J. & Yang J. Predicting human interruptibility with sensors: A Wizard of Oz feasibility study. In *Proc. CHI'03*, ACM Press (2003).
20. Isaacs, E., Walendowski, A., Whittaker, S., Schiano, D.J. and Kamm, C. The Character, Functions, and Styles of Instant Messaging in the Workplace. In *Proc. CSCW'02*, ACM Press (2002), 11-20.
21. Kohavi, R. and John, G.H. Wrappers for Feature Subset Selection. *Artificial Intelligence*, 97, 1-2, ACM Press (1997), 273-324.
22. Lotus Sametime http://www.lotus.com/sametime
23. Milewski, A.E. and Smith, T.M. Providing Presence Cues to Telephone Users. In Proc. CSCW'00, ACM Press (2000).
24. Nagel K. S., Hudson J. M. & Abowd G. D. Predictors of availability in home life context-mediated communication. In *Proc. CSCW'04*, ACM Press (2004).
25. Nardi, B., Whittaker, S. and Bradner, E. Interaction and Outeraction: Instant Messaging in Action. In *Proc. CSCW'00*. ACM Press (2000), 79-88.
26. O'Conaill, B., and Frohlich, D. Timespace in the workplace: dealing with interruptions. In *Conference companion CHI'95*, ACM Press (1995), 262-263.
27. Voida, A., Newstetter, W. C., and Mynatt, E. D. When conventions collide: the tensions of instant messaging attributed. In *Proc. CHI'02*, ACM Press (2002).
28. Witten, I.H. and Frank, E. Data Mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kaufmann (2005).