Inferring User Availability in Collaborations via Instant Messaging Activity

Munmun De Choudhury[1] **Ajita John**[2] **Dorée Duncan Seligmann**[2] **Hari Sundaram**[1]
[1]Arts Media & Engineering, Arizona State University [2]Collaborative Applications Research, Avaya Labs
Email: {munmun.dechoudhury,hari.sundaram}@asu.edu, {ajita,doree}@avaya.com

INTRODUCTION

This paper presents a framework that helps in automatically detecting the appropriateness of interruption for user communications in collaborative environments e.g. enterprises. Enterprise communications over the past few years have moved over from direct face-to-face interactions to interactions supported by a host of diverse multimedia technologies like video, audio and text devices. As enterprises go more 'social', a popular mode of communication that has emerged is instant messaging, popularly known as IM. IM based communication is useful in a modern enterprise because events in business processes such as a customer contacting customer service, the deadline for a task, or an application parameter crossing a threshold may trigger the need for a group of people to collaborate. Some of the example domains in which collaborations are common could be group tasks, peer-to-peer systems, conferencing, tele-medical applications and contact centers. Under these circumstances, IM based communication enables enterprise workers to be geographically distributed and yet engage in communication about these collaborative activities.

However, note that each of these domains often needs to cater to time critical communication context efficiently in order to satisfy the customers. In these complex communication routing scenarios involving enterprise workers who have varying expertise, are subject to different policies as a part of a global workforce and who might belong to different time zones, manually inferring the availability of a worker at a certain point in time (for an IM communication) might increase the latency in routing as well as make the entire communication process intractable. Besides, lack of awareness about the availability of workers might also result in unwanted, interrupting communication. Disruptive consequences of interruptions could be more mistakes; workers might have difficulties remembering and therefore hesitate and delay in making decisions, and also in general are likely to be less effective when exposed to interruptions. It is important to note here that the user-specified 'status' on an instant messaging client is not necessarily reflective of her *actual* availability for an incoming communication request.

Our Approach

There are two major contributions in this paper:

1. A proof-of-concept system that uses a dynamic Bayesian network framework to infer availability of users automatically.
2. A user study experiment to evaluate the application inferred availabilities.

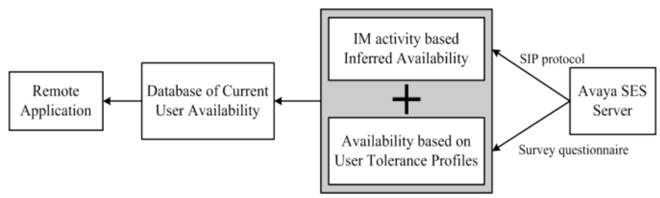


Figure 1: A schematic system diagram to infer availability of enterprise workers. Availability measures of users are computed dynamically for incoming communication requests based on (a) communication cues collected from IM logs of the users (for this, the SIP protocol is used to subscribe to the Avaya SES Server), and (b) user tolerance profiles of IM activity collected through a survey questionnaire. The composite measure is written to a database and can be read by a remote application which intends to get current availability measures of users.

We consider the communication context in a modern enterprise where instant messaging is used extensively by the enterprise workers for several collaborative tasks. Assume that the communication middleware in the enterprise (e.g. Hermes [?]) has a monitoring capability to capture several communication cues (or parameters) that characterize each user's IM activity – without compromising on her privacy. In this paper, we used the monitoring capability of Avaya SES server [?] and implemented a model to infer availability of users based on their activity with the Avaya SIP Soft-phone instant messaging client. A schematic system diagram is shown in Figure 1.

In this application, we model the monitored communication cues in a dynamic Bayesian network which yields posterior

This work was done while the first author was visiting Avaya Labs.

probabilities of user availability dynamically. We further use personalized tolerance profiles of the users to scale our computed measure – the tolerance profiles being collected through a survey in the enterprise beforehand.

In order to validate our inferred measures we conducted a user study with a total of eight users – four interns and four non-interns in a research lab setting (the Collaborative Applications lab at Avaya). The group was involved in discussion about a research paper of their interest for an hour each day over a week. Based on explicit feedbacks from the users about their acceptability of chat sessions as well as a post-experiment questionnaire, we attempted to understand the choices personal preferences made by the users and thereafter made comparisons of availabilities for each user with our inferred measure. Our findings show that our system can reasonably infer the availability of enterprise workers based on their instant messaging activity.

Related Work

Applications to detect the availability of users under various contexts have been of considerable interest in the past. Horvitz et al. in [2] developed an automated presence and availability forecasting service named COORDINATE. In a different work, Horvitz and Apacible in [3] present methods for inferring the cost of interrupting users based on multiple streams of events including information generated by interactions with computing devices, visual and acoustical analyses, and data drawn from online calendars. Gievska et al. in [1] present a framework that helps in selecting the most appropriate timing for interruption as a way to mediate human interruptions by the computer. Terveen at al. in [4] develop a model of social matching systems. Social matching systems bring people together in both physical and online spaces.

There are several limitations of prior work. The diversity, scale and intricacies of large modern enterprises present many challenges to the contextual reasoning of availability in middleware. Hence we present a framework which can exploit a host of communication context cues in enterprise user communications and use them to predict the availability of users. In this paper, we particularly focus on a communication media which is getting increasingly popular in enterprise domains recently – instant messaging activity.

The rest of the paper is organized as follows. Section 2 discusses our inference framework while in section 3 we present the methodology of a user study that was performed to evaluate our framework. We discuss the experimental results in section 4 and conclude in section 5 with the major contributions.

AVAILABILITY INFERENCE

We infer availability of a user based on two ideas: (a) application inferred availability using her IM activity, and (b) feedback based tolerance profiles of the user with respect to acceptance of IM communication.

DBN Framework

We discuss a dynamic Bayesian network based representation of parameters characterizing a user's IM activity. The DBN is then used to infer the probability of the user's availability, i.e. the posterior distribution of availability given the evidence over the different parameters.

We assume that the communication middleware in the enterprise has a monitoring capability to capture several communication cues (or parameters) that characterize each user's IM activity – without compromising on her privacy. The parameters in interest are: number of active chats at a certain point in time, degree of involvement in a chat (through number and size of messages sent and received), idle time since last message was sent, closeness of a chat session to completion and number of other users involved in the chat conference. We model each of these parameters as a probability distribution over a set of model parameters.

In a more general case, the model parameters might need to be estimated; however in this paper the probability of the communication cues given their ancestors are know explicitly from the IM activity logs collected from the monitoring client (based on subscription to the Avaya SES Server).

Tolerance Profiles

It is important to note here that the perception of availability of a user is extremely subjective to the user herself. In order to account for this subjectivity in this paper, we develop a notion of tolerance levels of availability per user, involved in the user study. The tolerance profile is a user's own perception of her availability based on her instant messaging activity. It is formally defined as the degree to which instant messaging activity on the user's part (both sending and receiving messages) is acceptable to her so that she is not interrupted in the existing work or communication. The parameters that can give a user an idea of her tolerance profile might be the number of chats she can handle or the number of instant messages she can send or receive.

Before the experiment is started off, we collect an initial feedback from the users in the group about their respective tolerance profiles. They are derived out of a questionnaire session before in which the profiles are as follows: (1) Not at all tolerant / MinTolerance, (2) LowTolerance, (3) MediumTolerance, (4) HighTolerance, and (5) Always fine / MaxTolerance.

These tolerance levels are coupled with the application inferred availability measures: High Availability (8-10), Medium Availability (4-7), Low Availability (1-3), Unavailable (0)] to modify (increase/decrease) the effective availability values using a suitable scaling factor. This effective availability is compared with the user mentioned interruption response to validate and verify the soundness of our measure. The closer the two interpretations are about the user's availability, the sounder would be our inferred availability status.

USER STUDY METHODOLOGY

We design an experimental methodology to validate the analysis that focuses on inferring the availability of a group of users in an enterprise. We construct an experimental setting in an enterprise (Avaya Labs, in this paper) where users would engage in considerable instant messaging activity to discuss items of technical interest / usefulness to them. Using explicit feedbacks from the users about their availability during such discussions, we can compare our application inferred measure to determine the effectiveness of our framework.

Experimental Setup

In this sub-section we discuss the setup for this user study experiment. A technical paper discussion bench is created in which the prime mode of communication is through the SIP Soft-phone instant messaging. A user set comprising enterprise workers belong to two loosely cohesive groups – summer interns (at the Collaborative Applications Research Lab, Avaya) and non-interns (at the same lab) is also created. Our main idea was that, along the course of a week (five business days), each day the group would hold a discussion through instant messaging to analyze the research paper on board for that day. This would construct the premise of the experiment. The paper which would be discussed is typically of any one user's research area of interest and is made know to the group a day earlier so that everyone can have a first hand view of it before the discussion actually commences.

The typical duration of the discussion is fixed at about an hour or so. This duration is further divided into four time slices, each 15 minutes. For the chat conversations to take place, we pair an intern, say I_1 with a non-intern, say N_1. Note, we always pair users belong to the two different sets, the interns and the non-interns, to ensure that there is effective discussion between them with regard to the research paper on board. Besides, the chat conversations in the experiments are designed to comprise exactly two users – they are not conference chats. However, each user could involve herself in multiple chat sessions at the same point in time based on system requests in the experiment.

Experimental Approach

We now present the experimental approach. Each time slice of the experiment comprises three kinds of message exchange – system-user exchanged messages, user-system exchanged messages and user-user exchanged messages. A schematic diagram of such message exchange is shown in Figure 2.

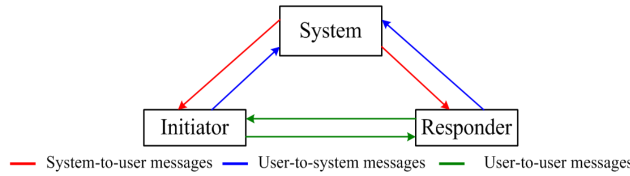


Figure 2: The message-exchange process in the user study experiment. The system picks an initiator requesting her preference to accept a new chat. Based on her response, the system contacts the responder and thereafter promotes a chat conversation between the two users.

We describe the message-exchange process in the experiment as follows:

1. The system picks an initiator, either intern or non-intern say I_1, and then sends a message to her, asking about her willingness to take up a chat conversation with a receiver N_1. The participant (I_1) might or might not choose to respond. If I_1 would choose to respond, we map her responses to a quantitative willingness scale from 0 to 10. The initiator could reply with any of the following options (experiment-specified): (a) *Yes, fine with this chat* (8-10 willingness) (b) *It is okay* (4-7 willingness) (c) *It's not very desirable* (1-3 willingness) and (d) '*No, not ready for this chat*' (0 willingness). We also conjecture if she does not respond, it means she is *Unavailable* or has a willingness of 0.
2. Should I_1 respond to the system, the system would send out another similar message to N_1 if she is willing to take this chat. If she does not respond, it means the user is Unavailable. If N_1 would choose to respond, she would have the similar set of options to rate her availability as in (1).
3. Once both the initiator and the receiver choose to chat based on their responses to the system messages, they are let go ahead and start a conversation involving discussion on the research paper on board for that particular day.
4. At the end of each time slice, a random pooling of interns (on every odd time slice, 1st and 3rd) or a random pooling of non-interns (on every even time slice, 2nd and 4th) is done to select users whom would be initiators of a new chat. The same process of message exchange is followed for all subsequent time slices and other days of the week.

Note to ensure privacy of the users, the content of the chats was not made a part of our experiments as well as the actions of one user were not revealed to others at any time during the experiment. Based on the user-mentioned availabilities at each time slice, we compare the corresponding application inferred measures for each user (that incorporates the tolerance profiles) and evaluate our framework.

Post-experiment Questionnaire

In order to understand and justify the explicit choices made by the users during the experiment, we used a post-experiment questionnaire to evaluate several factors related to the user choices and the experiment itself. The questionnaire comprised the following questions:

1. How much familiar the user was with instant messaging prior to the experiment.
2. If the tolerance profile changed after the experiment.
3. How well the act of accepting a new chat conformed to the actual activity level of the user in every time slice.
4. If the users had any external interruptions during experiment.
5. What were the degree and the nature of cognitive load experienced by the user during the experiment.

In the following section we discuss the experimental results comprising an analysis of availabilities determined by our method, and then evaluation of our framework based on the user study experiment.

EXPERIMENTAL RESULTS

In this section we discuss the experimental results. First we present some quantitative analysis of user availabilities based on the user experiment. Then we present evaluation based on the user specified availabilities, and finally an analysis of the user study responses.

Quantitative Analysis

We show inferred availability for all the participants in the user study in Table 1(a-d), over four different time slices of the experiment.

Table 1(a): Analysis of Inferred Availability (time slice 1).

Participant id	Inferred Availability	#Chats	#Messages Sent	#Messages Received	Size of Messages Sent	Size of Messages Received
P_8	0.09	1	8	4	438	343
P_9	0.03	1	13	10	562	486
P_4	0.002	2	14	21	829	1000
P_{10}	0.06	1	11	7	406	328
P_5	0.06	1	7	11	328	406
P_3	0.23	1	2	4	342	137
P_7	0.23	1	4	2	137	342
P_6	-	-	-	-	-	-
P_1	-	-	-	-	-	-
P_2	-	-	-	-	-	-

Table 1(b): Analysis of Inferred Availability (time slice 2).

Participant id	Inferred Availability	#Chats	#Messages Sent	#Messages Received	Size of Messages Sent	Size of Messages Received
P_8	0.0021	3	15	24	1394	1599
P_9	0.0063	2	25	12	1640	916
P_4	0.2709	2	8	11	291	201
P_{10}	0.2504	2	11	8	234	355
P_5	0.9805	2	5	3	170	133
P_3	0.0096	3	14	12	1433	1385
P_7	0.0088	2	12	18	1419	1380
P_6	0.6917	2	3	3	294	559
P_1	0.1285	1	10	8	701	921
P_2	0.4579	1	5	9	312	439

Table 1(c): Analysis of Inferred Availability (time slice 3).

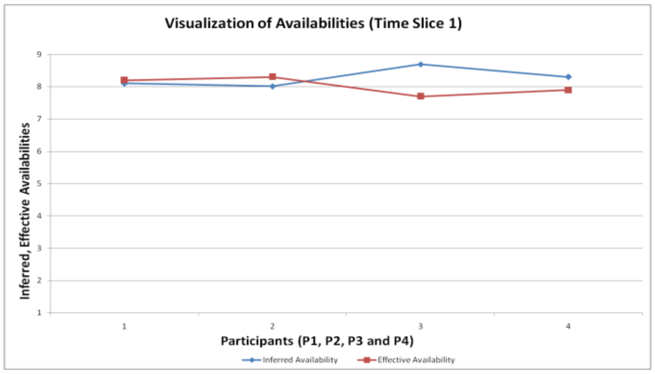
Participant id	Inferred Availability	#Chats	#Messages Sent	#Messages Received	Size of Messages Sent	Size of Messages Received
P_8	0.0141	2	10	14	988	1412
P_9	0.0081	3	14	9	1419	1331
P_4	0.2458	2	6	4	176	171
P_{10}	0.5706	1	3	4	89	80
P_5	0.1444	1	5	5	670	578
P_3	0.0015	4	10	20	1284	2153
P_7	0.1486	1	9	2	820	311
P_6	0.1136	1	6	5	860	423
P_1	0.4307	1	2	2	453	300
P_2	-	-	-	-	-	-

Table 1(d): Analysis of Inferred Availability (time slice 4).

Participant id	Inferred Availability	#Chats	#Messages Sent	#Messages Received	Size of Messages Sent	Size of Messages Received
P_8	0.0287	2	13	8	841	457
P_9	0.0015	3	21	15	792	1630
P_4	0.0095	2	11	16	931	867
P_{10}	0.0001	3	32	19	1574	2009
P_5	0.0013	2	25	23	971	759
P_3	0.0001	4	26	38	2114	1621
P_7	0.0128	2	9	17	1042	563
P_6	0.0807	1	12	9	352	343
P_1	0.0429	1	8	12	457	838
P_2	0.8616	1	1	1	54	41

Evaluation

Now we analyze the inferred and effective availabilities of the users based on our model in Figure 3.



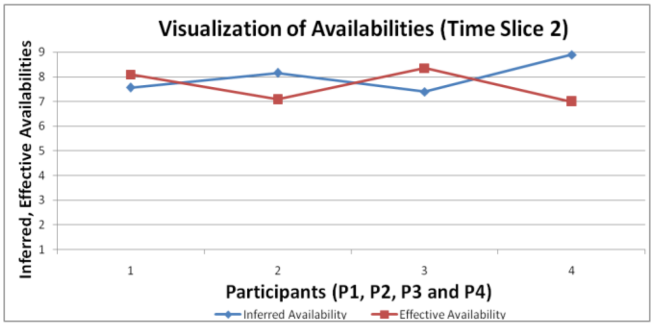


Figure 3: Inferred Vs. Effective availabilities for four users.

A histogram comparison is realized showing the accuracy/difference between the two availabilities. We have high availability prediction accuracy (~70%). The difference may be accounted owing to the personal tolerance of each participant. The results are shown in Figure 3.

User-Study Analysis

The following table in Table 2 present the net availabilities of the users based on their tolerance profile and the responses from the experiment.

Table 2: User-study analysis of availabilities.

Participants	Effective Availability (Time Slice 1)	Effective Availability (Time Slice 2)	Effective Availability (Time Slice 3)	Effective Availability (Time Slice 4)	Mean Effective Availability
P_1	0.001	0.0009	0.0001	0	0.0005
P_2	0	0.004	0.01	0.009	0.0057
P_3	0.46	0.69	0.44	0.01	0.39
P_4	0.06	0.12	0.58	0.04	0.2
P_5	0	0.47	0	0.12	0.16
P_6	0.08	0.005	0.008	0.001	0.02
P_7	0.36	0.003	0.13	0.002	0.07
P_8	0.9	0.5	0.27	0.002	0.38
P_9	0.06	0.5	0.11	0.86	0.41
P_{10}	0	0.11	0.13	0.028	0.11

CONCLUSIONS

In this section we present the conclusions of this work. We have developed a probabilistic framework to automatically detect the actual presence of a person (who is a domain expert in a particular area) and direct a communication, or merge the person in to an existing communication as and when the need arises in the field. The decision is made more critical under the time and resource constraints laid down in a virtual communication environment as of today and fits in well with the enterprise communication middleware Hermes developed at Avaya Inc. The communication baseline that is assumed to be used by all users (and pretty widely) is instant messaging (through SIP Soft phone clients). Results indicate that our model yields high prediction of use availability (~70%). This is further validated with the help of a small user study done in Avaya.

There are several interesting directions to future work. We are interested in validating how our model and results are likely to generalize to other settings and tasks. We are also interested in using machine learning techniques to automatically code availability from text and apply our techniques to our large corpus of dialogues from different media conditions.

REFERENCES

[1] S. GIEVSKA and J. SIBERT (2005). *Using task context variables for selecting the best timing for interrupting users*. Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies. Grenoble, France, ACM: 171-176.

[2] E. HORVITZ, P. KOCH, C. M. KADIE, et al. (2002). *Coordinate: Probabilistic Forecasting of Presence and Availability*, Proceedings of the Eighteenth Conference on Uncertainty and Artificial Intelligence, Edmonton, Canada, ACM, 224-233,

[3] E. HORVITZ and J. APACIBLE (2003). *Learning and reasoning about interruption*. Proceedings of the 5th international conference on Multimodal interfaces. Vancouver, British Columbia, Canada, ACM: 20-27.

[4] L. TERVEEN and D. W. MCDONALD (2005). *Social matching: A framework and research agenda*. ACM Trans. Comput.-Hum. Interact. 12(3): 401-434.