Enhancing Message Collaboration through Predictive Modeling of User Behavior

Biswajyoti Pal
Avaya Labs Research
211, Mt. Airy Rd
Basking Ridge, New Jersey 07920
Email:palb@avaya.com

Anupama Pasumarthy*
Dept. of Information Technology
Sreenidhi Institute of Science and Technology
Hyderabad, India
Email: anupama.pasumarthy@gmail.com

Krishna Kishore Dhara
and Venkatesh Krishnaswamy
Avaya Labs Research
211 Mt. Airy Rd, Basking Ridge, NJ 07920
kishoredhara@gmail.com, venky@avaya.com

***Abstract*—Research studies have shown that the effectiveness of collaboration and the choice of communication modality is intricately linked with the perceived presence and availability of the collaborating parties. Most collaboration systems offer users the ability to publish their presence for effective collaboration. However, a close observation of users' behavioral data shows a divergence such as in a published 'busy' state a user is actually willing to collaborate with certain people or in a published 'available' state a user is unwilling to collaborate with certain people. This behavior makes the notion of presence in collaboration systems ineffectual and often unreliable. In this paper, we propose a new predictive model of behavioral presence for collaborative messaging systems that automatically infers multiple presence states based on users expected collaboration behavior towards a contact. We present a novel confirmatory data mining technique that overlays a 'cluster of interest' on standard clustering techniques such as k-means, fuzzy k-means, and consensus clustering. We present validation results of our predictive model on data obtained from real-world deployed enterprise servers across multiple locations over a period of seven months.**

***Keywords*—Collaborative messaging systems, behavioral presence, and predictive modeling of user behavior**

I. Introduction

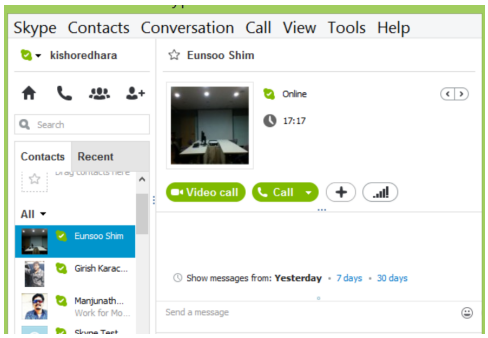
Unified collaboration clients that offer multiple modalites of communication such as audio, video, messaging, screen sharing etc., have become increasingly popular both in consumer domain and in enterprise or workplaces. Collaboration clients such as Microsoft Office Communicator, IBM Sametime, Skype, etc are popular in workplaces and clients such as Yahoo, Skype, Google Hangouts, etc., are popular among consumers. Two distinct features that can be observed from these collaboration clients are the increasing reliance on messaging as a primary or first step for collaboration and the wide usage of presence to indicate availability to various contacts. Messaging offers a near real-time and more personal collaboration than email. The role of presence in these systems is to enable effective collaboration in terms of expectations on the collaboration and in terms of interrupting other users. Several research studies [1], [2], [3] have demonstrated that the collaborating patterns of users change based on the context of interruption and the perception of availability. To mitigate this, several mechanisms are proposed to infer a user's exact presence and availability [4], [2], [5], [6].

The basic idea of collaboration with presence is that when users publish their presence status, all the contacts who subscribe or who are friends of the users can see the status of the user. These contacts then can decide what to expect from a collaboration session with the user, if they can interrupt a user, or the preference of one modality over another for a collaboration session. However, even with the move towards accurately inferring a user's current presence status based on their current activity, there are several problems with this current approach. One such problem is the limitation of "one" published or inferred presence status for everyone. In real world usage, a user's availability towards a contact is not universal. That is, a user might want to have one presence status to be published to his or her boss while they may want to publish another status to their colleagues or someone they are not willing to collaborate at that moment. Another problem is the nature of users' collaboration behavior with respect to their published presence status. Either users forget to change their presence status or do not conform to their published status. That is, a user with a 'busy' published state actually interacts with certain contacts and a user who is 'available' or 'online' does not interact with certain contacts.

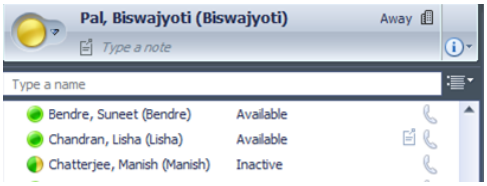
This divergence of behavior from the published presence status often makes the notion of presence ineffectual and any collaboration relying on presence either is interrupting users or is not guaranteeing expected levels of collaboration response. In this paper, we take a confirmatory data mining approach to capture a notion of behavioral presence that addresses these two issues. We build a predictive model that sets expectation of collaboration for a user's contacts, which is tailored towards each individual contact. We use enterprise users as our primary target for modeling as interruption management and effectiveness of collaboration is of primary importance in a workplace. The collection of data is followed by an initial exploration of data and data munging. We then build a model to predict user collaboration behavior through an iterative cycle of variable selection, learning, and validation. In each iteration, after clustering, we look at a 'cluster-of-interest', which given the feature variables of the model, essentially gives us how the behavior of a user deviated from the cluster that corresponds to their published presence status. We use a "closeness" function that identifies contacts in this 'cluster of interest' and change the published states for these contacts. These changed published states predict how users would respond to their contacts and also indicate whether contacts can interrupt a user. We use both internal validation and external validation to

*Research performed at Avaya Labs

COLLABORATECOM 2014, October 22-25, Miami, United States
Copyright © 2014 ICST
DOI 10.4108/icst.collaboratecom.2014.257317



(a) Skype Collaboration Client and presence



(b) MS Office Communicator and presence

Figure 1. Illustration of some collaboration clients that center around presence

evaluate our approach with real enterprise user data of over 20 plus users with several hundred contacts and close to a quarter million instances of collaboration messages.

The main research contributions of this paper are as follows.

1) For collaboration systems, we describe a new notion of behavioral presence that goes beyond one presence for all and increases the effectiveness of collaboration by bounding the expectations of users and matching the level of interruptability with the actual behavior.
2) We build a predictive model based on messaging systems and validate it based on actual data from users in work places across different locations.
3) We define a novel 'cluster-of-interest' overlay that can predict the actual effectiveness of collaboration based on a user's historical behavior towards each contact.

In the next section we describe in detail the problem of linking effective collaboration with perceived presence status and overview of our approach followed by related work in Section III. Section IV presents details of our data collection and Section V presents various phases of our algorithm. Finally in Section VI we present details of our evaluation followed by conclusions.

II. Effective Collaboration

In both workplaces and social situations, choosing the right communication medium that can minimize both the response times and interruption is crucial for effective collaboration. To facilitate this many modern communication systems such as Skype, Lync, Google, etc., offer the ability for contacts to see each other's presence. Figure 1 shows how various workplace and consumer collaboration clients use presence as an integral part of their clients. The idea is that the presence status, which is published by a user, indicates the willingness of a user to participate in a collaboration session and hence sets expectation of participants in terms of the response times and the effectiveness of collaboration. In addition to this, with unified communications, most messaging clients offer instant messaging, audio, and video collaboration, and choosing the right communication medium based on the presence status is crucial. When someone is on a call, sending a message may be is appropriate and may cause least interruption. Hence for effective collaboration appropriate notions of presence and availability are crucial.

However, there is a disconnect between existing presence systems and their usage in collaboration. The following list highlights the disconnect and how it affects collaboration.

1) **Stale State:** Users often choose the default state or forget to actively change their state to reflect their availability. This presence status is seen by all their contacts and their collaboration decisions depend on this stale presence status. Extensive work has been done that integrates status from communication servers and collaboration servers to infer states such as "in a meeting", "on a call", etc. While these increase the reliability of published presence status to drive collaboration and set expected behavior, often these inferences do not reflect a user's availability accurately because users may be available on other modalities during a meeting.
2) **One Status for All Collaborations:** Collaboration behavior of a user is often dictated by the participants in the collaboration. A simple example is a "busy" worker may immediately reply when there is a collaboration request from his or her supervisor and may completely ignore collaboration requests from certain contacts. This disconnect is caused by the inflexibility of one published presence status.
3) **Inconsistent Collaboration Behavior:** Even in cases when the published presence is as intended, users' behavior may not be consistent with this state. This inconsistency in collaboration is from both a user point of view and from the contact point view as well. For example, a "busy" user is willing to collaborate and an "available" user does not respond for collaboration requests.
4) **Inaccurate interpretations:** Often contacts of a user are intuitively aware of some of the above factors and start interpreting the published presence status in a way that fits them. These different interpretations of what a user would do in a presence status by their contacts will further increase the unreliability of the collaboration sessions. That is, some contacts may interpret a "busy" as available and send a collaboration request anyway or some users may not want to collaborate in a "busy" state or in a "in a meeting" state even though a user may be willing to engage in a collaboration session in perhaps a much less intrusive communication modality such as messaging.

In this paper, we look at the problem of effective collaboration from factors that influence the start of a collaboration, the expectation during a collaboration session, and excessive

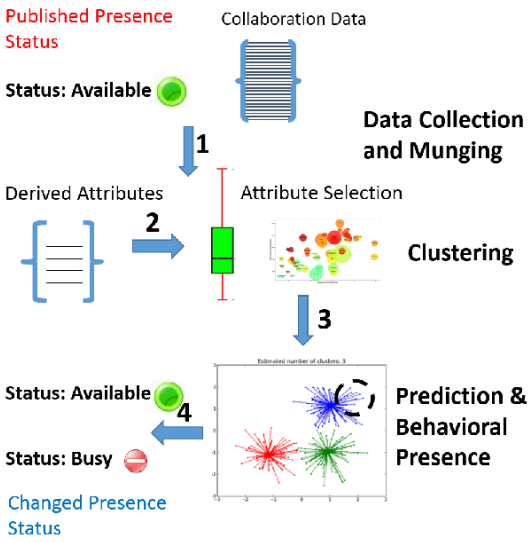


Figure 2. Overview of Behavioral Presence Computation

interruptions when users stop trusting the presence status and initiate collaboration requests.

Overview of our approach

We propose a new notion of presence based on a predictive model of user collaboration behavior. We term this notion as *behavioral presence*. Given a published presence status, p, of a user for all their contacts, behavioral presence is the modified presence status, p_b for certain contacts for whom the behavior of the user does not conform to the published status p but matches closely with p_b. Because behavioral presence is based on actual user behavior towards a contact, it minimizes or matches the intrusiveness factor expected by a user's published presence status and sets expectations for a participant in terms of the responsiveness of their collaboration session. In terms of computation, behavioral presence uses a predictive model to select a user's contacts for whom the published presence status is not reflective of the user's behavior towards them. It predicts a presence status for each of them that fits the user's behavior towards them.

The basic idea is to capture users' actual collaboration session data along with their presence states and build a predictive model that can capture how a user behaves towards each of his or her contacts. Given a published presence status of a user, which is viewed by all his or her contacts (see top left of Figure 2), our predictive model computes behavioral presence and changes the presences status for a subset of the user contacts (see bottom left of Figure 2). Figure 2 presents an overview of the process of building a predictive model and computing behavioral presence for individual contacts.

At high level, we can view various steps in Figure 2 in three phases. One is the data collection and data munging. The second step is to model a user's behavior in terms of the several feature variables which include observed variables such as presence status, message times, and derived variables such as mean response time of a session, variance of the response times, abandoned sessions, etc. These attributes are used to cluster similar behavior in a given state. The third step is an overlay of the cluster of interest (denoted by dashed circle in the figure), which uses a similarity factor to determine which of the contacts experience a surprising behavior in the published presence status. That is, in a "busy" state we look for a cluster of points that is closer to "available" behavior and change their behavioral presence to "available". This ensures that these contacts can effectively collaborate without thinking of interrupting the user and also guarantees a certain mean collaboration behavior even if the normal published presence is "busy". Note that for users who do not change their presence status often but change their behavior, this model computes a behavioral presence status that captures user's behavior towards each individual contact.

We evaluated our approach in a real enterprise across various locations with active users using MOC as their primary messaging system. We collected data for over six months and used both an internal validation for determining the number of clusters and external validation to determine the accuracy of our behavioral presence predictions. In the rest of this paper we give details of our approach and present our results.

III. Related Work

The importance of instant messaging as a means of collaboration has been well studied both in the consumer space and in workplaces [7], [8], [9]. In [7], Grinter et al., discuss the importance of instant messaging among teen users and how different consumer groups in the teen demography optimize messaging for various tasks. Isaacs et al [8] argue that messaging is used for mainly for work related tasks in enterprises. Nardi et al., [9] show how messaging supports many informal collaboration tasks and its effectiveness.

While the above studies show that the utility of messaging, studies such as [10] show how interruptions through messaging can adversely affect the performance of users. These effects include both responded interruptions and ignored interruptions. The contents of interruptions and their effects on mobile users are studied in [11]. Another aspect for effective collaboration is the response time, which is indicated by the presence status. In [12], the authors introduce a notion called "butter lies" where collaboration users invent "lies" as a way to avoid or explain unwanted interruptions, delayed responses, or long messaging sessions. Teevan [1] discusses how the projected presence notion affects users' communication.

In [13], Avrahami et al., build a statistical model for predicting response times of an instant message. Our work differs from their work in many ways. One is our notion of behavioral presence that tries to change the expected presence status instead of predicting a value in terms of 1, 2, 5, or 10 minutes for response times. Further, behavioral presence captures not just response times but also the actual presence status and a users' behavior towards all their contacts. That is, our system learns the behavior of a slow responding user from a fast responding user and does not try to quantify in terms of minutes. Finally, the data obtained in their study is from graduate students at the Carnegie-Mellon's HCI Institute and our data is obtained from real enterprise users going about their business across different locations. We believe that the data for our study is a better reflection of enterprise messaging behavior than a controlled environment.

Xu et al., [14] developed a model to estimate affective presence status and communication among experienced users. Their study is based on data obtained from an experience model, that is., users during the course of their normal work, pause and answer questions about their experience. Our work is different in both its capture of real user data and in not just trying to address what is an affective state but linking a user's presence status with their actual behavior. In [15], Pielot et al., analyze two weeks of mobile user data to build a model that uses seven features of their phone to predict if a user views a message in the next few minutes.

We focus on enterprise users, collect large volume of data for over six months. and address the effective collaboration issue from multiple dimensions.

IV. Message Collaboration Data

Our research primarily focuses on collaboration in workplaces for the following reasons. Unlike in the consumer space, in enterprise or workplace, minimizing unwanted interruption and enhancing the effectiveness of collaboration is crucial. Another reason for focussing on enterprises is that immediacy or expectation on response time is important in enterprises to resolve issues, get questions answered, or for other workplace tasks. Further, in workplaces, messaging is increasingly replacing calls as a primary means of collaboration. Also, enterprises enhance their collaboration systems by integrating with other communication servers to better reflect the user's status to their contacts, such as changing an "available" status automatically to "in a meeting" if the user is in a conference call.

Please note that our study does not focus on pre-scheduled meetings or collaboration as the user actions are agreed upon before and are often deterministic. Instead, we focus on collaborations that are on-demand and hence rely heavily on user behavior in various published presence states.

A. Data Collection

Due to privacy concerns, it is generally very hard to convince IT administrators to share data across large enterprises for a research study. Hence, instead of accessing data from collaboration or messaging servers directly, we had to develop a client software and seek volunteers that allow our software to monitor their messaging and upload their messaging data to a server. We developed an adjunct software to Microsoft Office Communicator (MOC) for capturing the data. Even for the volunteer users, to protect their data, our service level agreement with them is to capture only meta data of their collaboration and not actual contents.

B. Data Munging for Analysis

In the following examples and in the rest of the paper, we distinguish between a `user` and his or her `contacts`. The data captured is for each volunteer `user's` interaction with their contacts. The published presence status of the user is what the contact would be seeing and the published state of the contact is what the user would be seeing in his collaboration client.

The following steps illustrate the quantative aspects of various predictor variables used in our analysis.

user%BUSY	101	· · · 10:53 AM	contact	contact%ONLINE	6
user%BUSY	101	· · · 10:53 AM	user	contact%ONLINE	4
· · ·					
user%BUSY	101	· · · 10:54 AM	contact	contact%ONLINE	25
user%BUSY	101	· · · 12:53 AM	contact	contact%ONLINE	10

TABLE I. Truncated raw data from users

1) **Raw Data:**
Table I shows the format of the raw data uploaded from the client to the server. Each row in the table represents a message sent in a conversation from either user to contact or from contact to user. The second column gives the user id and the last column gives the length of the message. The fourth column captures the initiator of that particular message. Note that, the granularity of the capture, because of restriction in MOC interface, is a minute.
2) **Parameterizing presence states and direction of message:**

2	101	· · · 10:53:00	6	FALSE	contact	10
2	101	· · · 10:53:08	7	TRUE	contact	10

TABLE II. Parametrizing and Time normalization

Each row in Table II is a transformation of a corresponding row in Table I with the presence status represented as an integer (for example, 4 for "BUSY" and 2 for "ONLINE"), and TRUE for message initiated by user and FALSE otherwise. One additional transformation in this step is to normalize the time based on the number of messages exchanged in a minute.
3) **Inter-message time and Session Parameters:** The individual interactions from Table II is combined as a session. This merging allows us to perform a session based analysis. While it is sometimes easy to define a session in other forms of collaboration, such as telephony, it is not so intuitive for instant messaging and hence, we use a time based approach for determining sessions. If the inter message time

2	101	· · · 10:53:00	6	FALSE	contact	10	-1	1
2	101	· · · 10:53:08	7	TRUE	contact	10	8	1
2	101	· · · 10:53:16	33	FALSE	contact	10	8	1

TABLE III. Inter-message Time

between two instant messages is greater than a defined time duration, the former instant message shall mark the end of an instant messaging session and subsequently the later shall mark the start of an instant messaging session. This defined time duration is assumed to be 5 minutes for the purpose of further analysis. Table III gives inter message times for each of the message. The negative number in the first row indicates that there is no prior message for computing inter-message time.
4) **Session parameters and other derived attributes:** For every session, various parameters, such as the mean and variance of the inter message time, users

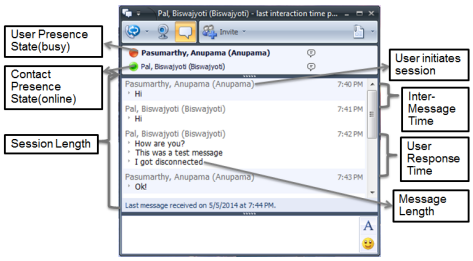


Figure 3. Derived attributes of an actual messaging session

User	Contact	Presence Status
<user>	<contact>	Online (10)

μ_{imt}	σ_{imt}	len	change	user-init	μ_{resp}
13.5	27.7	9	6	TRUE	15.3

$\sigma_{user-resp}$	$\mu_{contact-resp}$	$\sigma_{var-contact}$	last response
56.33	13	19	<time>

TABLE IV. A split table of derived attributes of a collaboration session for analysis that correspond to Figure 3

response time and contacts response time, whether that particular session is initiated by user, number of messages in the session, number of times the direction of the messages changed, the presence states during the session, and other such session parameters are computed. These parameters are later used for two purposes. First, to derive attributes of interest for a user-contact pair for the given user presence status and secondly, to enable a session-by-session validation of the predicted user behavior.

Parameters for all the sessions of a user-contact pair, for a given presence status, are transformed to obtain derived attributes such as mean of inter message time, users response time, contacts response time, ratio of the sessions initiated by the user, total number of sessions, completed sessions, abandoned sessions, etc. for each user contact pair.

Table IV shows this in a tabular form and Figure 3 relates the derived attributes of a session with an actual messaging session.

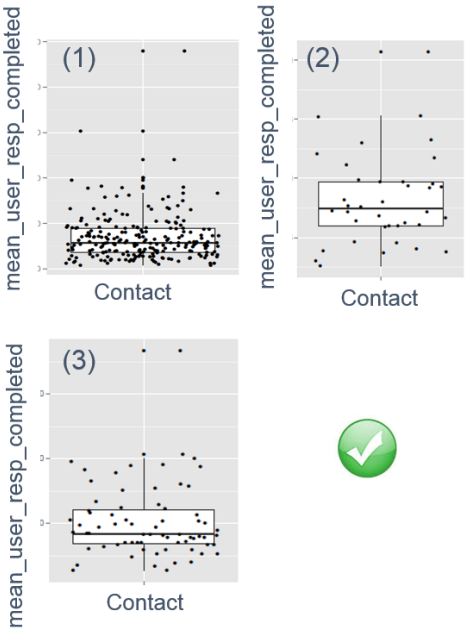
V. Modeling User Behavior and Behavioral Presence

In this section we first present our exploration of attributes for modeling user behavior, clustering user-contact messaging behavior, and present our algorithm for determining behavioral presence for collaborating users.

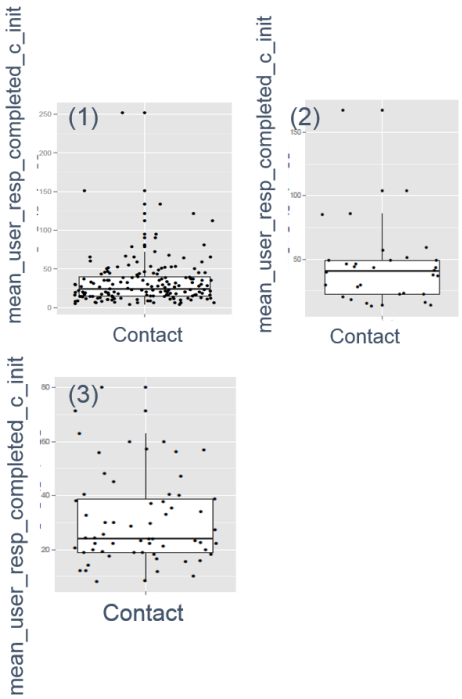
A. Selecting Attributes for User Behavior

Though we looked at many attributes, in this section, we describe results from our focus on three attributes that we used for modeling user behavior.

1) Response Times: Response times, especially response from a user to a contact, define user behavior towards a contact. This attribute not only satisfies and sets a sense of expectation but also is used by contacts in determining whether they can interrupt a user in any given state or not. Hence, response times



(a) For all sessions

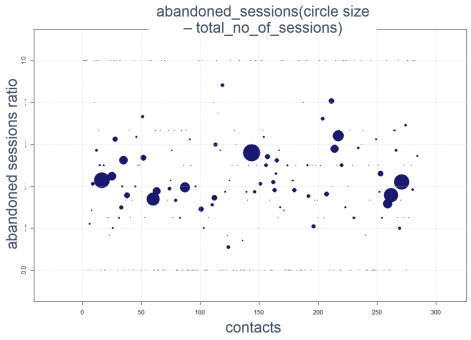


(b) For contact initiated sessions

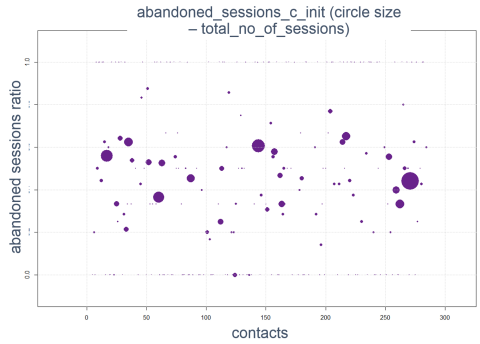
Figure 4. Sample response times distribution for three users.

is an important attribute for modeling user behavior towards his/her contacts.

Figure 4 shows the distribution of various response times. The left hand side of Figure 4 shows the distribution of mean user response times in all sessions for three sample users and the right hand side of Figure 4 shows the distribution of mean user response time in only sessions that are initiated by the



(a) All Sessions



(b) Contact Initiated Sessions

Figure 5. Ratio of Abandoned Sessions. Larger bubbles indicate larger sample size of the ratio.

contact. The similarity in these two distributions show that the mean response times over a session does not depend on who initiated the session. Please note that the distribution for user (3) appears different because of the scale of y-axis but follows the similarity pattern like the other users. Based on this data exploration and on the fact that even in user initiated sessions the response times of a user sets the expectation of a contact, we select mean user response time derived from all collaboration sessions as one of the model variable.

2) Ratio of Abandoned Sessions: While the response times capture user behavior in sessions that have at least one or more responses, the abandoned sessions also capture user behavior. Abandoned sessions are sessions either initiated by the user or by the contact that do not have any response. However, the expectation of a contact depends not on whether a particular session is abandoned but on the ratio of abandoned sessions. Figures 5a and 5b show the similarity of the ratio of the abandoned sessions in all the sessions and in contact initiated sessions. We select the ratio of abandoned sessions for both user and contact initiated sessions.

3) User or Contact Initiated?: Finally, we select the attribute we omitted from the above distributions, viz., whether a session is initiated by the user or by the contact. The intuition is to select an attribute that models user behavior from a contact point-of-view. A low user initiated ratio means contacts initiated more sessions than the user and a high user initiated ratio implies that users initiated more sessions than a contact. From a contact's point of view, a low user initiated ratio allows the model to see if the user behaves according to his or her published state and captures the expectation and the

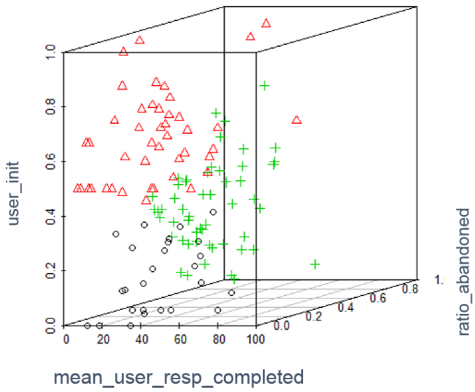


Figure 6. Clustering of contacts based on user behavior

interruption decisions from a contact's point-of-view.

B. Clustering User Behavior and Clustering Overlay

We use unsupervised learning with the selected attributes discussed in the previous section as the feature vector for capturing user behavior. We use standard clustering algorithms such as k-means clustering (and later we compare it with fuzzy c-means) for our clustering. The idea is that for each user, these clusters represent a group of contacts with whom the user had a similar collaboration behavior. Figure 6 shows a sample clustering of contacts for a user with the three attributes selected for our modeling. The circular dots indicate contacts that experience a quick response times, low ratio abandoned, and low user init (that is a high of contact initiated). The cluster denoted by the 'plus' sign indicates user behavior towards contacts that has high response times with high abandoned ratio and with low user init ratios. Similarly, the cluster indicated by the triangles denote user behavior that has low response times, high user initiated ratio, and medium to high abandoned ratio.

C. Cluster of Interest Overlay

In Figure 6, if the user's published state is "online" or "available", then for at least some contacts in the 'plus' cluster (or for points towards the right and top right), the user's behavior is unexpected because of high response times and high session abandoned ratios. This unexpected behavior results in ineffective collaboration in terms of contacts waiting for responses, waiting on abandoned sessions, and/or interrupting users.

Hence, for each presence status, we define a cluster of interest, which captures unexpected collaboration behavior with respect to a presence status. We use this cluster of interest as an overlay on the actual clusters to determine the contacts for whom published presence status does not match user behavior. We define this cluster of interest by its centroid in terms of the selected attributes.

The solid larger circle in Figure 7 indicates one such cluster of interest for the state "online". Note that we can have many clusters of interest. But in a published "online" state, we are interested in only the cluster where the behavior indicates that the user's behavior contradicts the "online" status.

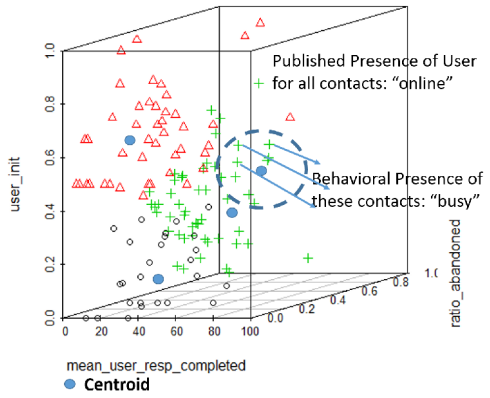


Figure 7. Relation between cluster of interest overlay and behavioral presence

State	Response Time	Abandoned Ratio	User init Ratio	Behavior is close to
Online	High	High	Low	Busy
Busy	Low	Low	Low	Online
		...		
In A meeting	High Low	High Low	Low Low	Busy Online

TABLE V. Picking expected centroid for the cluster of interest

Table V quantatively shows the possible values of the cluster of interest in various presence states. These three points indicated by response time, abandoned ratio, and user init ratio indicate the centroid of the cluster of interest. So for an "online" published state the cluster of interest is a state that contradicts the "online" status. That is, a cluster with centroid <high response time, high abandoned ratio, low user init ratio>. Similarly we arrive at other centroids for the clusters of interest in various presence states.

Note that while in states like "Online" and "Busy" there could be only one unexpected or extreme behavior, in states like "In a Meeting" the behavior could be like "Online", very responsive, or "Busy", unresponsive Hence, in some states after initial clustering we look at clusters of interest to see if there are any contacts that fall in this cluster for whom the published presence status does not match user collaboration behavior.

D. Behavioral Presence

In this section we present our algorithm to determine the behavioral presence of a user from the point of view of each of their contacts. The goal is to look at a user current presence status and determine the set of contacts for whom the behavior contradicts the published presence status. For these set of contacts, we predict the behavioral presence to be the state that is represented by the actual user behavior.

We can formulate this in terms of behavioral clusters and the cluster of interest discussed in the last two subsections. The set of contacts with contradictory user behavior is the intersection of one of the clusters with the cluster of interest. Figure 7 shows the cluster of interest and the intersecting points with one the clusters (with "plus" points). These points represent contacts who are expecting the published presence behavior but are experiencing contradictory behavior from the user. Hence, we compute these set of points and change the behavioral presence status for these contacts to match the behavior of the selected cluster of interest. In Figure 7, behavioral presence status would be "busy" for the contacts in the cluster of interest and in the "plus" cluster.

Even after computing various clusters of behavior and determining the cluster of interest for the user's presence state, we need to resolve two issues. One is to determine the cluster that is closest to the cluster of interest. The second is to determine the contacts in the intersection for whom the behavioral presence status should be changed. The second problem comes from the fact that often the clusters may not be well separated out from the cluster of interest and could be close. So we need a mechanism to determine that we are changing presence status of users only when there is a clear behavioral separation determined by the feature vector we selected.

The first problem is solved by a similarity measure. We normalize our feature vector and use Eucledian distance as a similarity measure. For the second problem we define a threshold with a goal that the threshold should guarantee that the all the points selected should be closer to the cluster of interest centroid than any other point that are in the other clusters. From the centroid of the cluster of interest, we calculate the minimum distance for all the clusters other than the most similar cluster. We take `th` to be greater than this minimum distance. Though this eliminates quite a few points from the cluster that is close to the cluster of interest, it guarantees that we are only changing presence status of contacts that are closer to the centroid of cluster of interest than any other point in the other clusters.

Bringing everything together, the input to our behavioral presence algorithm is the current presence status and the historical meta-data of the user's messaging behavior. The output of the behavioral presence algorithm is a list of contacts for whom the current published status has to be changed to a new behavioral status that is accurate in predicting the effectiveness of their collaboration at that point. We summarize our discussion in the following steps.

1) For each user u, collect all the data and compute session attributes
2) For each presence status p, compute mean user response time, abandoned session ratios, user session initiation ratios, and the expected centroid of cluster of interest c_e (as in Table V)
3) Compute i clusters c_i using k-means, fuzzy k-means, or consensus clustering
4) Pick the cluster of interest c_e^p for presence status p.
5) Let the presence status represented by the cluster of interest c_e^p be p_b
6) For each centroid c, compute similarity of c and c_e^p
7) Select the cluster that has highest similarity as the cluster of interest, say c_s.
8) Pick the threshold th as follows.
   a) for each of the c_i, where $c_i \neq c_s$, compute the minimum distance of a point in c_i to the centroid of the c_e^p. Let that be be represented by m_i

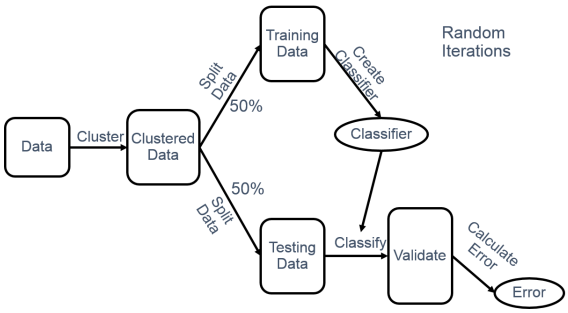


Figure 8. Internal Validation through replication analysis

b) threshold th = min($\vec{m}$).

9) Chose all the points in c_s with distance to the centroid of c_e^p that is less than the threshold.

10) Change the status of these contacts represented by points in c_s from p to p_b.

VI. Implementation and Evaluation

A. Implementation Details

We developed several components as part of our research study starting with data collection. For data collection, an adjunct to Microsoft Office Communicator was implemented using the Office Communicator API SDK to collect the meta-data related to every instant message and to upload the meta-data to a server. This data is collected on users' machines and periodically uploaded with an acknowledgement to ensure that data is properly uploaded to a server. Over a period of six months, from over 20 users and several hundred contacts, we collected meta-data on about quarter million user-contact interactions.

Periodically, the data collected on the server is downloaded for analysis, modeling, and for validation of our algorithm. The data pre-processing, derivation of attributes, clustering and validations internal and external validations were run in R.

B. Internal Validation through Replication Analysis

We performed replication analysis, which is described below, to evaluate the stability of clusters mined by our model, and to estimate the number of clusters. Figure 8 illustrates various steps in our replication analysis.

1) Cluster using the attributes identified in the algorithm in previous section
2) Repeat the following steps 20 times
   a) Randomly select about 50% of the clustered data without replacement as the training data and 50% as the testing data
   b) Create various classifiers using linear regression, regression tree, random forest, and k-NN on the training data using the attributes as the predictor variables and cluster as the outcome
   c) Use these classifiers on the 50% testing data

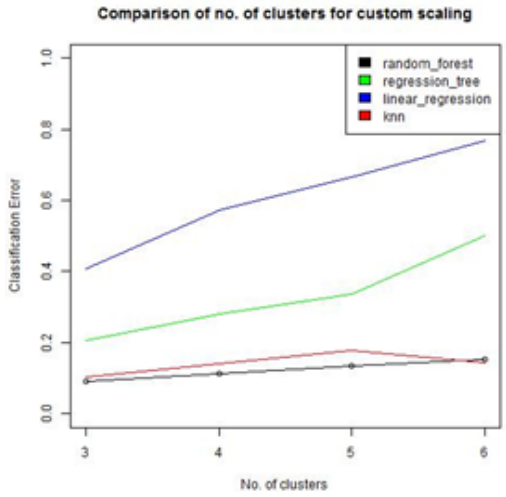


Figure 9. Internal Validation through replication analysis

d) Compute the classification error using the predicted cluster from the classifier and actual cluster obtained from clustering

3) Collate error rates from multiple runs

Figure 9 shows that the classification for different classifiers used with varying number of clusters. The classification error is lowest when the number of clusters is 3. This internal validation of the processed data through replication analysis shows that for our data there are three latent stable clusters. We use a cluster size of 3 for our data to predict behavioral presence. Note that, this number could change for a new set of users or even for the same users over a period of time. We need to perform the internal validation periodically and arrive at the number of clusters that is needed for the behavioral presence algorithm.

C. Behavioral Presence Results

User	Published Presence Status	% of contacts in cluster of interest	second cluster of interest
Ross	online	50	
Dhara	online	50	
Manoranjanhan	online	38.46	
Walani	online	35.71	
Shah	busy	50	
Vasudev	busy	50	
Kakade	busy	25	
Ezell	busy	20	
Pal	in a meeting	40	60
Ross	in a meeting	25	50
Ezell	in a meeting	59.09	4.55
Alfred	in a meeting	0	61.54

TABLE VI. User wise percentage of contacts that experienced a contradictory behavior and hence required behavioral presence status change

Table VI presents the perentage of unexpected collaboration behavior in terms of cluster of interest in that presence status. That is when user `Ross` is online, 50% percent of his contacts expect a high response time or high abandoned ratio and hence effectively their behavioral presence is "busy".

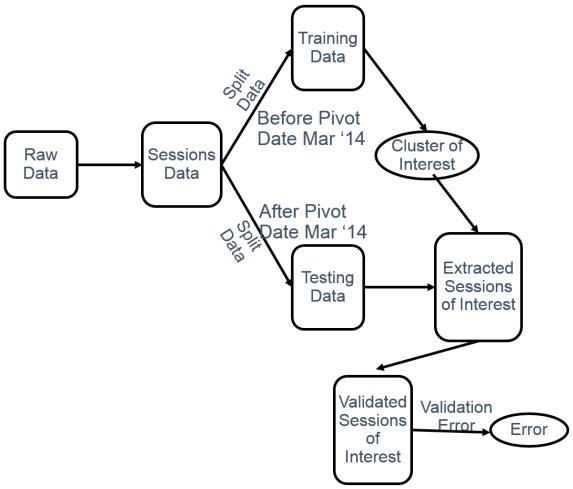


Figure 10. External Validation of Behavioral Presence

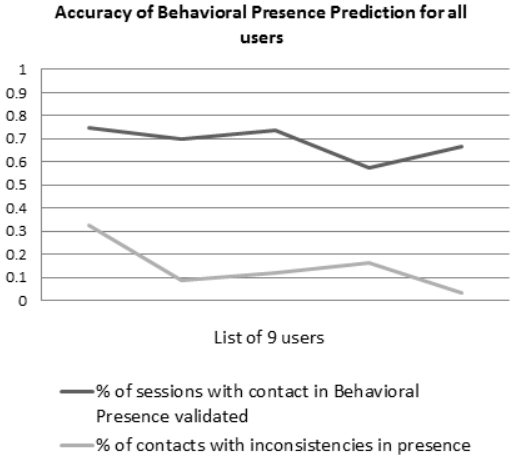
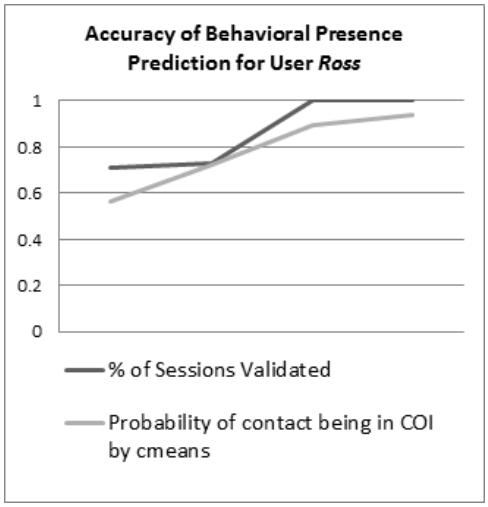
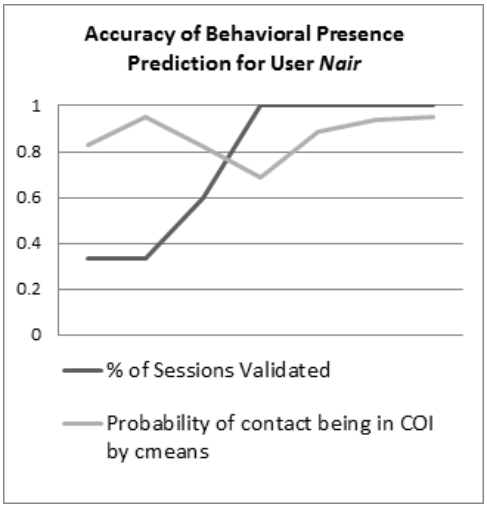


Figure 11. Accuracy results for all sessions of 9 users



(a)



(b)

Figure 12. Sample results that show the accuracy of behavioral presence for individual users

Similarly the table presents results for several users in different presence states. Note that for "in a meeting", we look for multiple clusters of interest, that is if a user is "available" or if a user is "busy" and compute the percentages of those clusters independently.

These results demonstrate that users' behavioral data supports our claim that collaboration users tend to behave differently towards their contacts. It further validates the need for individual presence status for their contacts and the notion of behavioral presence.

D. Accuracy of Behavioral Presence Prediction

Finally, to see if our model and behavioral presence accurately predicts the collaboration behavior of a user, we use external validation. Figure 10 shows various steps in external validation for computing the accuracy of behavioral presence and Figures 11 and 12 present results from our external validation. The basic idea is to take a pivot date in our data and use data prior to that pivot to build our model and predict behavioral presence. For each session after the pivot date, we validate each session if the predicted behavioral presence is accurate or not. For example, if the behavioral presence is changed from ONLINE to BUSY, the sessions after the pivot time are validated by observing users' behavior manually and marking them as conforming or not conforming. We define accuracy as the fraction of number of sessions that behavioral presence predicted accurately over the total number of sessions after the pivot date.

For evaluation, we use both fuzzy C-means and k-means for the initial clustering as mentiond in the previous section. While using K-means algorithm enables us to determine if the contact is within the cluster of interest, fuzzy C-means allows us to define degree of membership of the contact in the cluster of interest or rather the probability of confirmation of sessions with contacts with their respective behaviors. Thus the percentage of sessions validating follow the probability of behavioral confirmation. For certain contacts, our notion of behavioral presence is accurate in predicting the presence behavior even though the published presence status is different.

For example, for Ross in Figure 12a, for certain contacts the accuracy is 1 but in general the accuracy is quite satisfactory for all the users. We believe using certain derived attributes, such as time since last interaction, session length etc., as model variables will further increase this accuracy.

VII. Conclusions

In this paper, we look at an important problem of effective collaboration and how users and their contacts often rely on inefficient presence systems for collaboration. We present a new notion of behavioral presence that is based on data mining and on a predictive model of user behavior. Behavioral presence captures users' behavior towards each individual contact and predicts a status change if necessary so that contacts can set their expectations in terms of responsiveness or minimize their interruptions. In the behavioral presence algorithm, we use a cluster of interest overlay over standard clustering algorithms. Finally, we present several evaluation results using data obtained from real users on deployed enterprise networks across different locations.

Acknowledgment

The authors would like to thank all the volunteers who participated in our research study.

References

[1] J. Teevan and A. Hehmeyer, "Understanding how the projection of availability state impacts the reception incoming communication," in *CSCW*, A. Bruckman, S. Counts, C. Lampe, and L. G. Terveen, Eds. ACM, 2013, pp. 753–758.

[2] M. Czerwinski, E. Cutrell, and E. Horvitz, "Instant messaging and interruption: Influence of task type on performance," *OZCHI 2000 conference proceedings*, vol. 356, p. 361, 2000.

[3] E. S. De Guzman, M. Sharmin, and B. P. Bailey, "Should i call now? understanding what context is considered when deciding whether to initiate remote communication via mobile devices," in *Proceedings of Graphics Interface 2007*, ser. GI '07. New York, NY, USA: ACM, 2007, pp. 143–150. [Online]. Available: http://doi.acm.org/10.1145/1268517.1268542

[4] J. B. Begole, N. E. Matsakis, and J. C. Tang, "Lilsys: Sensing unavailability," in *Proceedings of the 2004 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '04. New York, NY, USA: ACM, 2004, pp. 511–514. [Online]. Available: http://doi.acm.org/10.1145/1031607.1031691

[5] M. Wiberg and S. Whittaker, "Managing availability: Supporting lightweight negotiations to handle interruptions," *ACM Trans. Comput.-Hum. Interact.*, vol. 12, no. 4, pp. 356–387, Dec. 2005. [Online]. Available: http://doi.acm.org/10.1145/1121112.1121114

[6] L. Dabbish and R. Kraut, "Controlling interruptions: Awareness displays and social motivation for coordination," in *In Proc of CSCW 2004, ACM Press*. ACM Press, 2004, pp. 182–191.

[7] R. E. Grinter and L. Palen, "Instant messaging in teen life," in *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '02. New York, NY, USA: ACM, 2002, pp. 21–30. [Online]. Available: http://doi.acm.org/10.1145/587078.587082

[8] E. Isaacs, A. Walendowski, S. Whittaker, D. J. Schiano, and C. Kamm, "The character, functions, and styles of instant messaging in the workplace," in *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '02. New York, NY, USA: ACM, 2002, pp. 11–20. [Online]. Available: http://doi.acm.org/10.1145/587078.587081

[9] B. A. Nardi, S. Whittaker, and E. Bradner, "Interaction and outeraction: Instant messaging in action," in *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '00. New York, NY, USA: ACM, 2000, pp. 79–88. [Online]. Available: http://doi.acm.org/10.1145/358916.358975

[10] M. Czerwinski, E. Cutrell, and E. Horvitz, "Instant messaging and interruption: Influence of task type on performance," 2000.

[11] J. E. Fischer, N. Yee, V. Bellotti, N. Good, S. Benford, and C. Greenhalgh, "Effects of content and time of delivery on receptivity to mobile interruptions," in *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services*, ser. MobileHCI '10. New York, NY, USA: ACM, 2010, pp. 103–112. [Online]. Available: http://doi.acm.org/10.1145/1851600.1851620

[12] J. Hancock, J. Birnholtz, N. Bazarova, J. Guillory, J. Perlin, and B. Amos, "Butler lies: Awareness, deception and design," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '09. New York, NY, USA: ACM, 2009, pp. 517–526. [Online]. Available: http://doi.acm.org/10.1145/1518701.1518782

[13] D. Avrahami and S. E. Hudson, "Responsiveness in instant messaging: Predictive models supporting inter-personal communication," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '06. New York, NY, USA: ACM, 2006, pp. 731–740. [Online]. Available: http://doi.acm.org/10.1145/1124772.1124881

[14] A. Xu, J. Biehl, E. Rieffel, T. Turner, and W. van Melle, "Learning how to feel again: Towards affective workplace presence and communication technologies," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '12. New York, NY, USA: ACM, 2012, pp. 839–848. [Online]. Available: http://doi.acm.org/10.1145/2207676.2208524

[15] M. Pielot, R. de Oliveira, H. Kwak, and N. Oliver, "Didn't you see my message?: Predicting attentiveness to mobile instant messages," in *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*, ser. CHI '14. New York, NY, USA: ACM, 2014, pp. 3319–3328. [Online]. Available: http://doi.acm.org/10.1145/2556288.2556973