Profile Sharing Recommendation System for Enterprise Collaboration

Shivangi Agarwal*
Dept. of Computer Science and Engineering
Indian Institute of Technology
Kharagpur, India 721302
Email: twsh517@gmail.com

Krishna Kishore Dhara and Venkatesh Krishnaswamy
Avaya Labs Research
211 Mt. Airy Rd
Basking RIdge, NJ 07920, USA
Email: {dhara, venky}@avaya.com

***Abstract*—Collaboration among enterprise users is often enhanced by their prior history that may include summary of the topics discussed, people involved, documents shared, and other such profile information. In enterprises, unlike in consumer collaboration, sharing this profile information could be extremely valuable and would further enhance the effectiveness of users in a collaboration session. For instance, a sales team member interacting with a potential customer over a period of time can share the profile or contextual information he or she gathered with a new member of the team. Sharing such profile information would jump start the collaboration of the new member and enhance the effectiveness of the collaboration. In this paper, we propose a session-based graph algorithm that recommends profile sharing possibilities to enterprise users based on the history of their collaboration strengths with persons interacting with them. We present results from implementing our algorithm on collaboration relationships that are computed on a deployed real-world enterprise server with approximately one million user-person relationships over a period of nine months.**

Keywords*—*Enterprise collaboration, contextual collaboration, user profiles, collaboration relationship strength

I. Introduction

Collaboration among users has been extensively studied both in consumer communication networks and in enterprise or business communication networks. While both these communication networks focus on collaboration across various modalities such as audio, video, and instant messaging, there are distinct differences in the evolution of collaboration in consumer networks and enterprise communication networks. Consumer networks have seen increased collaboration in online social networks (OSNs). Correspondingly there is extensive research that focuses on how information is diffused and shared among users in OSNs [1], [2], [3], [4].

On the other hand, collaboration in enterprise networks is largely focused on a seamless unified communication experience that allows users to navigate sessions through different modalities and to bringing the right people at the right time on the right device. Recent advances in enterprise networks include augmenting collaboration experience with contextual awareness such as location, sensory information, etc. While these enhance the collaboration among enterprise users and increase their productivity, they still are not linked with enterprise users' communication behavior, their history, and to the business process. In this work, we take collaboration among enterprise users beyond modalities of communication and sharing of documents to sharing of profiles based on their dynamic social behavior and current context. Our work aims to enhance enterprise user collaboration by providing them context that has been diffused through trusted parties and help integrate historical knowledge into collaboration sessions.

The following use case scenario illustrates how enterprise users can benefit from this enhanced collaboration. Consider a meeting between an enterprise sales team and some customers. While the modalities of communication and basic sharing allows them to share an audio/video bridge and some documents, the prior interaction between these parties is not shared. Further, if a user who is mediating cannot attend then the session often ends up with a rehash of what happened in a prior session. In this paper, we present a system that builds dynamic social relations among enterprise users and persons that they collaborate with, and recommends appropriate profile sharing to enhance collaborative sessions.

Though social relations among enterprise users seem like OSNs, there are fundamental differences between the two. The primary difference is enterprise relations are dynamic. That is, unlike OSN, the "friends" in enterprise networks are not selected by the user. Further, they change often, sometimes within a few hours. For example, based on geographical location, an enterprise user might work with different teams at different times. Users' enterprise social network can change during the course of a day and hence, highly temporal in nature. Another important difference is that information is private and highly sensitive in enterprises and one cannot use standard information diffusion principles to share. Further, the relations and the recommendations are based on each users' own data and not on a collective network.

In this paper, we propose a system that mines enterprise users' data and builds a personalized, temporal collaboration matrix that captures their collaboration strengths. Using this matrix, we build a profile sharing strength graph for each user that captures how profiles can be shared across a users' contacts. Finally, given the current context of the user, we do a constrained walk on the graph to recommend profiles that could be useful to the user. Our system has been tested on a real enterprise communication network server that has been running over an year with around 250 users and about 1 million user-person relationships.

*Research performed at Avaya Labs

COLLABORATECOM 2013, October 20-23, Austin, United States

DOI 10.4108/icst.collaboratecom.2013.254076

The main contributions of our work are as follows. We present a system that builds a dynamic collaboration relationship network for each individual enterprise user and uses this system to recommend profile sharing that enhances end-user collaboration. We use users' current context to recommend profile sharing in collaboration sessions. We present results based on a deployed enterprise server for over a year.

The rest of the paper is organized as follows. In Section II we present several enterprise use case scenarios and define our problem. Section III presents related work. In Section IV we discuss how to compute collaboration matrix and profile sharing graph. Section V presents out recommendation algorithm and discusses privacy policy overlay. Section VI presents our implementation details, results, and our evaluation followed by conclusions.

II. Profile Sharing and Contextual Awareness in Enterprise Collaboration

We present some use case examples in business collaboration applications to motivate the effectiveness of profile sharing among users. Later, in this section, we define the problem of profile sharing, its scope, and challenges.

A. *Profile Sharing: Use Case Examples*

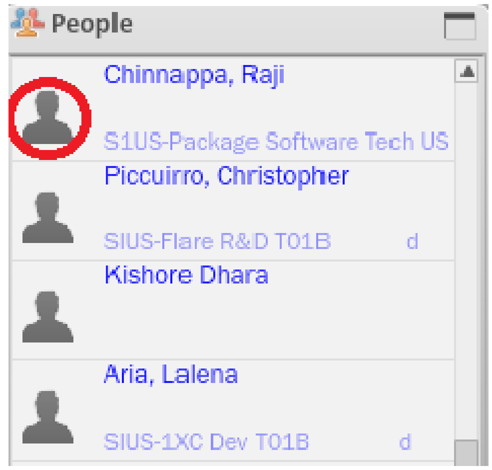
Since the scope of our work and deployment is in enterprise collaboration, we limit our use case examples to real-world enterprise scenarios. The examples below are only for illustration and do not represent an exhaustive list of the utility of profile sharing in enterprise collaboration systems.

New Team Memebers: Consider a team collaborating on a project or a sales deal over a period of time. Each user in the team brings a context or a profile that is built over a period of time that includes history of prior conversations, documents shared, and results of several meetings. In such a case, consider the case when a user, say A, is being replaced by another user B or if another user B is added to the team. Sharing profile of the participants of the prior collaboration sessions from A's perspective with B will enable B to understand the context of the collaboration sessions and contribute rapidly.

Away from the team: If a user is on a vacation and comes back then obtaining a shared profile from a colleague will help the user get a quick status update automatically.

Searching Relevant Profiles: Sharing profiles across enterprises based on some enterprise privacy policies will allow users to search the shared profiles. For example, typically attorneys in a medium or a large law firm are assigned to different cases and often similar cases are assigned to different attorneys. If the attorneys can search the profiles of others who are working on similar cases and understand the issues or challenges of the cases, it would expedite the handling of those cases. Note that the profile sharing here is not just searching for the cases, but for the meta data about people and their interaction in such collaboration sessions. Then combining personnel with similar cases or bringing a new case with shared profiles results in efficient collaboration between the law firm and its clients.

Sharing Picture Profiles: Finally, we discuss a picture profile



(a) Enterprise Contacts Application

(b) Enterprise Collaboration Session with four participants on an Avaya Flare IPad client

Figure 1. Enterprise collaboration applications with missing profile pictures

sharing example in enterprise. This example helps in illustrating a simple notion of profile and in solving a real problem in enterprise applications. Fig. 1a and Fig. 1b show typical instances of enterprise collaboration applications with missing pictures for several of a user's contacts. Unlike, online social networking sites like Facebook, Twitter, there is no central repository of pictures or profiles that users can upload and share in enterprise. Further, a more important distinction and the reason why this problem is non-trivial in enterprises all the contacts or persons collaboration with a user need not be part of the enterprise. Hence, enterprise applications cannot access their profile pictures or in general profiles from a central repository, even if such a repository exists. Hence, sharing picture profiles among themselves based on their collective connection strengths would enhance enterprise collaboration applications.

There are several reasons why we use sharing profile pictures as an illustrative example. First, the simplicity of sharing profile pictures will allow us to focus on the nuances of profile sharing in collaboration and on computation of profiles, which could vary from application to application. Though, in later sections, we discuss how the notion of profile is related to enterprise privacy policies, in this paper we focus on a generic notion of recommending profile sharing for collaboration that is applicable to any profile.

Secondly, this separation allows us to draw a clear distinction with our earlier work on computing contextual awareness for collaboration [5], [6], [7], [8], where we focus on computing context awareness that enhances enterprise collaboration applications. Focusing on the profile pictures and keeping the problem we are solving generic will allow us to present our work independently.

B. Challenges of Profile Sharing

The following list presents various challenges to profile sharing and highlights that profile sharing among enterprise collaboration users is a non-trivial problem.

1) **Relevance and Connection Strength:** A typical enterprise user during the course of his or her work activity collaborates individually or in groups with hundreds of people. Profiles or contextual information of all these users is computed and potentially can be shared with others. Sharing every profile a user has with other users will lead into problems of irrelevance, noise, and may even violate enterprise privacy. For instance context of a supervisor's discussion about an employee in his team with his superiors should not be shared with the employee. It is important, to find the relevance and the connections among users to determine who can share what profiles.
2) **Temporal Relationships:** Certain users are crucial only during collaboration sessions. Sharing profiles that are valid or relevant only during those sessions is essential. We need to determine how profiles can be shared when users, with low prior connection strength, are in a collaboration session. While temporal profiles capture transient session relationships, they do not capture the transient nature of the profiles, which we term as *dynamic profiles*.
3) **Dynamic Profiles:** Unlike contact information, profiles are dynamic and using an outdated profile would be incorrect. For example, a picture profile could be outdated or the profile of a user on the areas they are working, contributing, or their availability during a collaboration session could change over time. The dynamic nature of the profiles as well as the connection strengths of the users sharing the profile should be considered before sharing profiles.
4) **Privacy:** Finally, enterprises can enforce a "profile sharing policy" that needs to act as an overlay over the connection strengths of collaborating users and the profiles gathered for them.

C. Profile Sharing

In the rest of this paper, we use the term *user* to denote people who are provisioned on our system. That is, all the computations or periodic invocation of our algorithms are performed for these users. The term *person* is used to denote people that collaborate over an audio, video, conferencing system, IM, etc., with users of the system. These persons need not be provisioned in the system. Some users can be persons because they interact with other users of the system. That is, if user A has a conference session with user B, then for user B, user A will be a person. This allows us to treat everyone that interacts with B as persons. However, note that all persons are not users of the system. An external customer collaborating with a user in the system will a person but not a user of the system.

In its most generic sense, we define the profile sharing as follows. Let $U = \{u_1, \cdots, u_n\}$ be the set of enterprise users. For each user, $u \in U$, let $P = \{p_1, \cdots, p_n\}$ be the set of persons that collaborate with over a period of time using emails, conference calls, IMs, and calls. Let $Pr = \{pr_1, \cdots, pr_n\}$ be the corresponding set of profiles for persons in P. Let C_{u_1} be the current context of user u.

Profile Sharing Recommendation: For a user u, we define *profile sharing recommendation* as the set of tuples $(pr_k, \text{path}_k) \cdots, (pr_y, \text{path}_y)$, where pr_k are the profiles that can be shared and path_k is the sequence of persons through which the profile pr_k can be shared based on u's connection strength with $p_i \in P$, enterprise policy, and participants in the current context C_{u_i}.

We frame the sharing of profiles among collaborating peers as a recommendation problem to avoid pushing irrelevant profiles and to allow users to select only profiles that they are interested in. Note that when the current context C_{u_i} is not known or when user is not in any collaboration session, the default recommendation considers only the connection strength and the enterprise policy. In Section III we discuss how this notion of recommended set based on collaboration strength in the context of enterprise collaboration is fundamentally different from general consumer recommendation algorithms and OSN connection strengths.

D. Overview of the Solution

The basic idea of our solution is to look at the collaboration strengths across various communication and collaboration modalities such as email, IM, audio, video, and conference calls. We are not just interested in a user's connection strength with people interacting with him. Instead, given a user we compute a collaboration matrix of persons interacting with the user. That is, with respect to a user we compute a pair-wise collaboration score of persons interacting with that user. In essence, this gives not just the relationship a user has with persons he or she is collaborating with but how persons collaborating with him are related. Note, this is a crucial component for sharing among collaboration peers and is computed independently for every user in the system.

From the collaboration matrix, we then build a profile sharing graph based on the collaboration strengths. For each user, we update a profile table that keeps track of the profiles the user owns and can share and the profiles that the user can only consume. Using the profile sharing graph and profile table, a walk of the graph based on the entries of the profile table that can be shared we compute the recommended list of profiles for sharing. Finally, this recommended list is filtered using an enterprise privacy policy to get a final recommended list. If the recommendation is in the context of a collaboration session, the graph is pruned based on the participants of the collaboration session and a walk of the graph with the profile table will give the recommended list of profiles for sharing.

In later sections we discuss our algorithm(s) in detail and present results from implementing our algorithm on a deployed server on real enterprise servers with close to 1 million user-person relationships.

III. Related Work

User behavior in the context of online social networks (OSNs) and online recommender systems is quite different from enterprise user behavior. Our work is distinctly different in its scope, requirements, and approach. We limit our discussion of related work to these aspects.

In OSNs, the notion of *homophily*, people exhibiting similar characteristics tend to be connected [9], clustering, and information diffusion [3], [2] has been extensively studied in the literature. Granovetter [10] in his seminal paper presents the strength of weak ties in a social network. Another aspect is to find the importance of a node or the role it plays in propagating information across networks [1], [11]. Several OSN advertising and marketing applications use these algorithms to find related people for targeting advertisements. There is a subtle distinction in the way information diffusion is studied in OSNs and the way we study collaboration strength in enterprises. In OSNs, while the connections are explicit the information diffusion is implicit. That is, the connectivity is driven by users requesting a friend or accpeting a friend, and once the connectivity is established, the information diffusion depends on the OSN graph and policies. However, in enterprise collaboration environment the connectivity is implicit and the information diffusion is explicit. That is, enterprise users have no prior network structure with whom they interact and the information sharing collaboration sessions are explicitly driven by the user. These aspects render the large volume of research in OSN orthogonal to profile sharing and other aspects of enterprise collaboration.

Recommender or recommendation systems attempt to recommend information items that are likely to be of interest to the user. Some examples of the information items are TV programs, movies, music, news, books, web pages, etc. The commonly accepted formulation of the recommendation problem was first stated in [12], [13], [14] and this problem has been studied extensively since then. The recommender systems are classified into three categories based on how recommendations are made [15]: demographic-based filtering, collaborative filtering and content-based filtering. The main differences between context-based communication services and recommender services are the scope of the learned model (one user versus all users), lack of any user input, and the dynamic and real-time nature of the content.

Guy [16] uses social media behavioral data to recommend people that have similar interests and OSN behavior. While this may result in interesting recommendations of people similar to an enterprise user, we are interested in recommending people for profile sharing based on their collaboration strength within an enterprise. The people we recommend need not have similar interests or similar behavior outside of an enterprise, for example in OSNs.

In [17], Praveen et. al., describe a system, SBone, on how personal devices can share resources in a social network. Braghin et. al. [18] present ways to secure and share private information in social networks and others [19] who use attribute based cryptography to overlay a security layer on information sharing in OSNs. These and other such methods are orthogonal to our problem of finding or recommending profile sharing for enterprise users. Some of these encryption techniques could be used to ensure the privacy of shared profiles. That integration is beyond the scope of profile sharing problem discussed in this paper.

Our earlier work [8], [5], [20], [7] on contextual awareness in unified communications deals with various aspects of enhancing a user's unified communication experience by predicting relevant people, documents, conversation threads, and events. Our prior work does not deal with any notion of sharing profiles or sharing computed context across different users.

To the best of our knowledge, there is no prior research that finds a list of recommended profiles for enterprise collaboration users.

IV. Building Profile Sharing Graph from User Collaboration Behavior

In this section, first, we motivate the need to compute person-to-person collaboration relationship with respect to each user in the system. Then, we present our algorithm for profile sharing recommendation for default or no context and for in-context during a collaboration session.

A. Terminology and User Profile Privacy

We use U and u_i to denote the set of users and individual users respectively of our system. Similarly, we use P and p_i to denote the set of persons and an individual person collaborating with any user. Users are part of an enterprise that are subscribed to the profile or awareness computation engine. Persons are anyone who are interacting with any given user and hence can be from outside an enterprise. Note that, the computation of a person's profile is with respect to a user only. That is, we respect the privacy boundaries of users in enterprise and compute the profile of a person with respect to a user. With a different user, the same person can have a different profile computed based on their interactions.

Our earlier work [5], [20] presents detailed description of how various correlations across communication units, which we refer as *comunits*, such as email, calls, conferences, etc., are computed and used for relevance ranking. We omit details of our work on these except for computing person-to-person relationships because our work on profile sharing recommendation is independent of those computations.

B. Collaboration Matrix

To understand why we need person-to-person relationship with respect to each user instead of a simple user-to-person consider the following scenario. Often profile sharing among collaborating peers occurs through a lot of intermediate peers. While these peers may be strongly connected to the user, the profile of the person they can share may not be strongly connected with respect to this user. For example Bob and Alice collaborate quite a lot and Alice and Charlie collaborate a lot. However, Alice can share Charlie's profile only when their strength is high with respect to collaboration sessions with user Bob. Translating that to our notation, person Alice can

TABLE I. SAMPLE COLLABORATION MATRIX FROM OUR DEPLOYED SERVER BETWEEN PERSONS COLLABORATING WITH USER `VENKATESH`

	James	Harvey	Raji	...
Harvey	0.6555	45.66	45.78	...
Raji	50.89	5.70	69.54	...
Venkatesh	20.43	56.23	120.65	...
...	...	...	...	...

share person Charlie's profile with user Bob only when the connection strength of user Bob and person Alice is strong and the connection strength of person Alice is strong with person Charlie with respect to collaborations involving user Bob. So the person-to-person relationship of Alice and Charlie with respect to User Bob is important.

For each user, for each pair of persons, p_1, p_2, they collaborate with, we compute their relationship, R_{p_1,p_2}, for user u as follows. Note that, for clarity, we omit u superscript in our description.

$$R_{p_1,p_2} = \quad w_1 * Com_{p_1,p_2} + w_2 * Thread_{p_1,p_2} + w_3 * SR_{p_1,p_2}, \quad (1)$$

where w_1, w_2, w_3 are constants.

Com_{p_1,p_2} is the aggregated score of all *comunits* in which p_1, p_2 are participants along with user u. This score captures how relevant is p_2 to p_1 from u's from a connectivity point of view.

$Thread(p_1, p_2)$ is the aggregated score of the thread participation of p_1 and p_2. These threads could be email threads or conference call threads, etc. This score captures the active participation of p_2 when p_1 is in the conversation thread.

$SR(p_1, p_2)$ is the aggregated score of *comunits* where p_1 is directly sending to p_2 or vice versa with user u as one of the recipient.

For the constants w_1, w_2 and w_3, we collected data from users and ran regression tests to show the difference between default values and user preferences. We ran Multiple R, R Square, Adjusted R Square, Standard Error, and Observations to determine the values w_1, w_2, and w_3.

Table I shows a sample collaboration matrix for a user. Few things to note from the table.

1) Each user has a unique collaboration matrix where the values are computed based on the Equation 1. Tabel I shows the collaboration matrix for `Venkatesh`.
2) The rows and columns of the matrix represent persons interacting with the user.
3) Each element of the matrix is represented as the collaboration strength (based on Equation 1) with respect to the user. That is, for user `venkatesh`, the collaboration strength of person `Harvey` when `Raji` is present in the collaboration session is 5.70.
4) The row `venkatesh` represents user `venkatesh` as a participant in the conversation. Hence, that row represents a direct relationship between `venkatesh` and other persons collaborating with him.

Equation 1 indicates that the values of the collaboration matrix change with each new conversation such as a collaboration session, an emails, an IM, or a call.

C. Profile Sharing Graph Algorithm

1) Normalization and Decay: There are several crucial aspects to collaboration matrix described above. One is, for efficiency, only the affected elements of the matrix are computed. That is, the person-to-person value is computed only when there is a new collaboration session involving those two persons. This means that the values of $Comm_{p_1,p_2}$, $Thread_{p_1,p_2}$, and SR_{p_1,p_2} in Equation 1 are scores at the time of their active state or when there is new activity in that thread or with respect to the (p_1, p_2). If there is no other activity then the corresponding elements of the collaboration matrix are not modified. Hence, to find collaboration strength at any given instance, we need to decay the relationship strengths. Though this decay is needed to reflect the changing relationships for profile sharing, the real-time nature of the score for profile sharing is less sensitive for profile sharing. We use a relatively slow decaying function for profile sharing, which reflects that changing sharing profiles need not be a real-time as relationship strengths are built over a period of time. The current relationship strength, crs_i^j for row i and column j is based on value in the collaboration matrix, r_i^j, and the number of days, num, since the last collaboration session involving i and j.

$$crs_i^j \quad = \quad Max(r_i^j \times (1 - 0.1 \times log(1 + num)), 0), \quad (2)$$

Another aspect is the need for normalization of the collaboration strength score. The scores captured by collaboration matrix are cumulative and are directly proportional to the collaboration level of a user. A global threshold across all users to decide strong and weak relationship strengths for profile sharing does not reflect individual user behavior. That is, a global threshold may be too strong for low collaborating individuals and similarly, a low global threshold may be too weak for highly collaborating users. Hence, we use normalized threshold values on a per user basis to decide strong and weak strong strength for profile sharing. Another aspect for selecting a normalized threshold value is to choose a value for each row that captures both the relationship strengths across various rows and also isolates the strength in relation to each individual row. The following equation 3 captures these two aspects.

$$t_u^i \quad = \quad 1/n \Sigma_{j=1}^n \mu_j + \sigma_i, \quad (3)$$

where t_u^i is the threshold for a user u and for i in the collaboration matrix for u, μ_j is the mean of row j, n is the number of rows, and σ_i is the standard deviation of row i.

2) Building a Profile Sharing Graph: Based on the threshold t_u^i across all rows of a collaboration matrix and the decayed values, we build a profile sharing graph, P_G^u, for u. The nodes of P_G^u are persons in the collaboration matrix and for each (decayed) value $v_{i,j}$ in the matrix that is greater than t_u^i, there is a directed edge between j and i. Fig. 2 shows a sample profile sharing graph based on Table 1. Note the missing edges between `Venkatesh` and `James`, and `Raji`

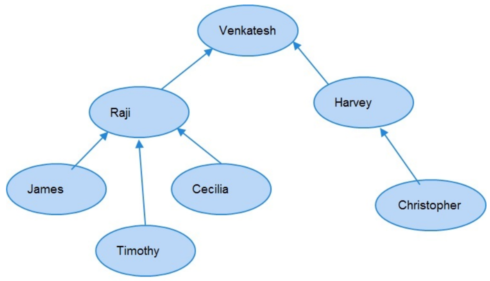


Figure 2. Sample Profile Graph for collaboration Matrix in Table 1

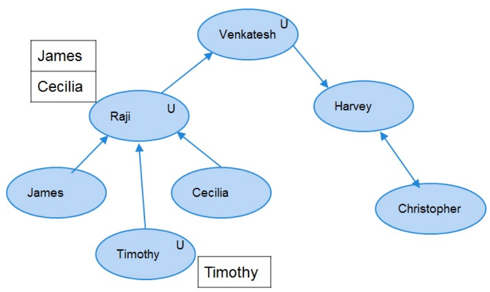


Figure 3. Profile Graph marked with Users configured in the system marked as “U” with their profile tables listed.

and `Harvey` because of the low values in the collaboration matrix. The arrows indicate the direction of profile sharing which is the reverse of the relative collaboration strength.

For each user, the corresponding profile graph, P_G^u changes with every collaboration session they participate in. Recomputing the entire collaboration matrix and correspondingly the entire profile graph for sharing is quite expensive. We take the 'modification only' approach to recomputing the matrix and the profile graph. That is, only rows that are affected with new collaboration sessions are affected in the matrix, and only corresponding nodes and edges are modified. Though we compute the mean and standard deviation in an incremental fashion, there will be stale edges that represent values above an older threshold. Based on Equation 2 and on incremental computation of threshold, we limit this staleness by recomputing the entire profile graph periodically, say once in a day for each user.

V. Profile Sharing Recommendation

Profile sharing recommendations are delivered to a user by walking the profile sharing graph in combination with two major aspects. The first one is user owned profiles and their privacy policies. The second one is the current context of the user. That is if the user is in the default state, no collaboration session, or in-context with a collaboration session.

A. *User Owned Profiles, Privacy Policy, and Validity of Profiles*

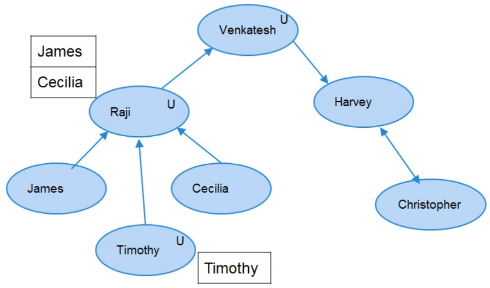
For clarity, in this paper so far, we have not made a distinction between the collaborations a user is having with various persons. That is, some of the persons collaborating with the user can be themselves users of the system. These users are tracked and often obtain other profiles from their own profile sharing recommendations. These obtained profiles are stored in a table called *profile table*. Clearly, only users of the system have these profile tables.

For profile sharing, users of the system have a distinct advantage because they can share not only their profiles but other profiles they have in their profile tables if the privacy policy is satisfied. To see this distinction consider the profile sharing graph in Fig. 3 for `Venkatesh` with users of our system marked as “U”. Since `Venkatesh` is the owner of the profile sharing graph, we are interested in the information of profiles we have for other users `Raji` and `Timothy`. Other nodes in the graph are persons that collaborated with `Venkatesh` along with the persons or users in the intermediate nodes. These persons can only share their profile but `Raji` can share her profile along with `James'` and `Cecilia's` profiles. Note that, this kind of sharing enables the use case scenarios, such as **new team members**, **away from the team**, etc., discussed in Section II.

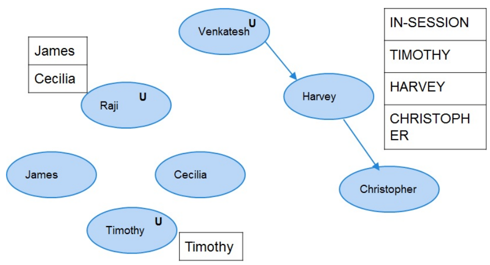
The privacy policy of profile sharing depends on the profile being shared and the nature of enterprise. For example, sharing a customer profile with a colleague may be allowed in some enterprises but in health and financial sectors such sharing may need to satisfy external regulatory policies, such as FCC in United States. In this paper, we limit our discussion to a privacy policy that allows domain specific policies as an overlay for the profile graph.

In the profile tables that are associated with each user, profiles are marked with different kinds of sharing attributes such as *public*, *private*, *shared*, *inferred*, etc. These attributes reflect the way the profiles are obtained and determine how these profiles are shared. For example, in Fig. 3, `Raji` can freely share `James'` profile with `Venkatesh` if the profile she has is *public*. On the other hand if it is *private* then either she cannot share the profile or needs `James'` permission to share the profile. *Shared* profiles are obtained from other users and *inferred* profiles are obtained from a user's collaboration behavior over a period of time. The sharing of these are often guided by enterprise policies and user choices. For the picture profile sharing example, since there is no `inferred` profile picture, we only use `public`, `private`, and `shared` attributes. For picture profiles with `shared` attribute, we allow a free sharing of the picture profile. This framework supports the basic mechanisms to overlay a domain specific privacy policy for sharing profiles using the profile graph.

Another attribute that is attached to the profiles is the validity of the profile. Once a picture is shared, this attribute indicates the period of time for which the shared profile is valid. After that the profile has to be deleted and when recommended by the engine, a new profile has to be obtained. For example, if the profile `James'` validity expires then `Raji` cannot offer that profile to be shared with `Venkatesh`. Further, she may choose to obtain a new profile of `James` when the system recommends `James'` profile to her.



(a) Profile Sharing Graph



(b) Default Mode

Figure 4. Collaboration Session mode with 3 participants

B. Profile Sharing Recommendation

Finally, for each user, a profile sharing recommendation algorithm looks at the profile sharing graph of the user and the user's current context to determine the set of person profiles that can shared. It recommends a set of tuples, that each contain a profile to be shared along with the path of sharing.

In the default case, with no current collaboration session, the following steps are performed for a user u with a profile graph PG. In the overview of our algorithm below, we use RS for final recommended set, Rec for intermediate recommended set, N for the set of nodes to be explored, and c, n for individual nodes.

1) Initialize the recommend set, Rec to $\emptyset$ and the set of nodes to be explored N as $\{u\}$.
2) For each $n \in N$
   a) $Rec \longleftarrow Rec \bigcup \{(p, (c, n, \cdots, u)) \mid Edge(c, n) \in PG \wedge \text{profile } p \in c\}$
   b) $N \longleftarrow N \setminus \{n\} \bigcup \{c \mid Edge(c, n) \in PG\}$
3) $RS \longleftarrow filter(Rec)$, where for each $(p, (c, \ldots, u)) \in R$, the function *filter* uses p's privacy attributes and enterprise privacy policies to decide if p can be shared or not.
4) RS is the recommended profile-path sharing tuple that is presented to the user u.

The step 2 a) above collects all the profiles and paths reachable from a node and step b) adds news paths that need to be explored in the profile graph. In 2 b), for clarity, we omit adding path along with nodes that need to be explored in the loop (step 2). Step 3 filters the profiles that can be shared based on the policies as discussed in the previous section.

The profile recommendation set RS above is for default case, that is when there is no collaboration session. In the case of a collaboration session, the computation described above remains same but the profile graph that is the input to the recommendation algorithm is pruned based on the participants of the session. Fig. 4a and Fig. 4b highlight the difference between profile sharing in default and in a collaboration session. In a session between `Venkatesh, Timothy, Harvey, and Christopher`, the recommendation algorithm ignores the edges with nodes that belong to persons that are not in the session. In this case, the only recommended profiles are of that `Harvey` and `Christopher`. However, if `Raji` were to be added to the session, the recommended profile set will include `Raji, James, Cecilia,` and `Timothy`.

TABLE II. SUMMARY OF DEFAULT PROFILE SHARING RECOMMENDATION FOR `VENKATESH`

Level 1	Level 2	Level 3	Level 4	Level 5
70	468	1	0	0

TABLE III. DEFAULT PROFILE SHARING RECOMMENDATION FOR SOME SAMPLE USERS IN OUR SYSTEM

Name	Level 1	Level 2	Level 3	Level 4	Level 5
John B	41	141	28	0	0
Sophie G	138	257	27	5	1
Elizibeth H	99	586	97	11	1

VI. Implementation Details, Results, and Evaluation

We have implemented our algorithms for computing collaboration matrix, profile sharing graph, and recommendation on real enterprise users across several countries. This deployed server has around 250 users with close to a 1 million user-person relationships over a period of nine months. To avoid skewing up of our observations we have enterprise users ranging from senior executives, directors, researchers, developers, pre-sales, and sales teams. Further these users are spread acorss North America (mostly USA and Canada), Brazil, and across many locations of Europe and Asia.

We use the template in Table II to describe results of profile sharing recommendation. While the actual results include the profile and path to the profile, we use notion of *Levels* to describe profile sharing for a user's collaboration. Level 1 indicates that the profile sharing is within in a node distance of 1 in the profile sharing graph. Similarly, for Level 2, Level 3, etc the node distance with respect to the user will be 2, 3, etc respectively. Table II shows that the number of profiles recommended for `Venkatesh` with a path length of 1 is 70 and with a path length 2 is 468. For example, a sample Level 1 is `Venkatesh` $\longleftarrow$ `Raji` and a sample Level 3 is `Venkatesh` $\longleftarrow$ `Raji` $\longleftarrow$ `Sarang` $\longleftarrow$ `Sandra Wong`.

Table III lists the profile sharing recommendation for some users and Table IV gives the profile sharing recommendation summary for all users in our system. From our results we observed several things. First, in our implementation, for real-time performance the algorithm for profile recommendation is terminated when the path length is 5. Looking at the results

TABLE IV. DEFAULT PROFILE SHARING RECOMMENDATION FOR ALL USERS IN OUR SYSTEM

Level 1	Level 2	Level 3	Level 4	Level 5
6787	11345	657	77	22

and the number of recommendations, our assumption seem to not miss out too many profile sharing recommendations. Note that, while our implementation may miss out on some long path recommendations, our algorithm presented in earlier section does not suffer from this problem.

A common thing we observed is the higher number of Level 1 through 4 profile sharing recommendations. Intuitively, this indicates that quite a few enterprise users have strong collaboration strengths with groups of 3 or 4. Another interesting aspect we saw was that role of an enterprise user seem to reflect in the number of profile sharing recommendations. For example, Elizibeth, who is in **pre-sales** has more LEvel 3, 4, and 5 connections than other enterprise users such as research director, project manager, developer, etc.

Finally, the overall profile sharing results listed in Table IV show that the profile sharing recommendations are considerably smaller than close to 1 million user-person relationships in our system. This is indicative of the high threshold values used for collaboration strength matrix to profile sharing graph conversion. A stronger enterprise privacy policy or a weaker enterprise privacy policy could strengthen or weaken these thresholds.

Evaluating our algorithm analytically or measuring its performance against an expected result set is quite hard in an enterprise collaboration networks. The data privacy requirements are much more stringent and even though our server looks at various communication elements, we guarantee to our users that we do not store any information. Further, computations are limited to a user and correlations across users cannot be done because of enterprise privacy policies.

The two approaches to measure the efficacy of our algorithm is to deploy it and get a feedback from users. Another way is to develop clients that can push direct feedback from clients on a) share of recommendations they are accepting, b) propagation of the shared profiles, and c) usage of profiles. We selected some users of our trail, akin to user trials, to do a deep dive on the recommendations and the corresponding paths for sharing profiles. Though not quantitative, we found their responses to be quite satisfactory. On the second approach, we built a client to share picture profiles. We need to integrate it with clients of our users' choice to get a holistic feedback.

VII. CONCLUSIONS

In this paper we looked at a novel mechanism to enhance collaboration experience for enterprise users. We present several real-world use cases to illustrate the novelty and motivate the need of integrating profile sharing in collaboration. We based our approach on the behavior of enterprise users and people collaboration with them to build a collaboration matrix, a profile sharing graph, and a recommendation algorithm for recommending shared profiles. Our system has been tested on deployed enterprise communication systems with real enterprise users with close 1 million user-person relationships.

ACKNOWLEDGMENT

We would like to thank hundreds of users of our system and especially grateful to the users who were willing to go over their data, relate and provie feedback to the recommendations from the algorithms presented in this paper.

REFERENCES

[1] P. Domingos and M. Richardson, "Mining the network value of customers," in *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM, 2001, pp. 57–66.

[2] D. Gruhl, D. Liben-Nowell, R. Guha, and A. Tomkins, "Information diffusion through blogspace," *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 2, pp. 43–52, 2004.

[3] G. Kossinets, J. Kleinberg, and D. Watts, "The structure of information pathways in a social communication network," in *KDD '08: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM, 2008.

[4] S. Strogatz, *Sync: The emerging science of spontaneous order*. New York, NY, USA: Hyperion., 2003.

[5] K. K. Dhara, V. Krishnaswamy, E. Shim, and X. Wu, "Enhancing unified communication services with communication context," *JCM*, vol. 7, no. 2, pp. 115–131, 2012.

[6] K. K. Dhara, V. Krishnaswamy, and P. Katre, "Inferring relevant people from multi-modal user communication activity," in *CCNC*. IEEE, 2012, pp. 538–543.

[7] E. Shim, K. K. Dhara, X. Wu, and V. Krishnaswamy, "Communication data based user activity recommendataions." in *proceedings of ACM ICUIMC*. ACM, Feb. 2011.

[8] K. K. Dhara and V. Krishnaswamy, "Integrating multiple contexts in real-time collaboration applications," in *Proceedings of the 3rd Workshop on Context-awareness in Retrieval and Recommendation*, ser. CaRR '13. New York, NY, USA: ACM, 2013, pp. 7–14.

[9] M. McPherson, L. Smith-Lovin, and J. Cook, "Birds of a feather: Homophily in social networks," *Annual Review of Sociology*, vol. 27, pp. 415–444, 2001.

[10] M. S. Granovetter, "The strength of weak ties," *American Journal of Sociology*, vol. 78, pp. 1360–1380, 1973.

[11] D. Kempe, "Maximizing the spread of influence through a social network," in *In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003, pp. 137–146.

[12] W. Hill, L. Stead, M. Rosenstein, and G. Furnas, "Recommending and evaluating choices in a virtual community of use," in *Conference on Human Factors in Computing Systems*, 1995.

[13] P. Resnickm, N. Iakoyou, M. Susha, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *Computer Supported Cooperative Work Conference*, 1994.

[14] U. Shardanand and P. Maes, "Social information filtering: Algorithms for automating 'word of mouth'," in *Conference on Human Factors in Computing systems*, 1995.

[15] M. Balabanovic and Y. Shoham, "Fab: Content-based collaborative recommendation," *Comm. ACM*, vol. 40, no. 3, pp. 66–72, 1997.

[16] I. Guy, S. Ur, I. Ronen, A. Perer, and M. Jacovi, "Do you want to know?: recommending strangers in the enterprise," in *Proceedings of the ACM 2011 conference on Computer supported cooperative work*, ser. CSCW '11. New York, NY, USA: ACM, 2011, pp. 285–294.

[17] P. Shankar, B. Nath, L. Iftode, V. Ananthanarayanan, and L. Han, "Sbone: Personal device sharing using social networks," 2010.

[18] S. Braghin, V. Iovino, G. Persiano, and A. Trombetta, "Secure and policy-private resource sharing in an online social network." in *SocialCom/PASSAT*. IEEE, 2011, pp. 872–875.

[19] R. Baden, A. Bender, N. Spring, B. Bhattacharjee, and D. Starin, "Persona: an online social network with user-defined privacy," in *Proceedings of the ACM SIGCOMM 2009 conference on Data communication*, ser. SIGCOMM '09. New York, NY, USA: ACM, 2009.

[20] X. Wu and V. Krishnaswamy, "Advanced communication thread analysis and applications." in *ICC*. IEEE, 2012, pp. 1971–1975.