Probabilistic Paths and Centrality in Time

Joseph J. Pfeiffer, III
Purdue University
Department of Computer Science
West Lafayette, IN 47907
jpfeiffer@purdue.edu

Jennifer Neville
Purdue University
Department of Computer Science
West Lafayette, IN 47907
neville@cs.purdue.edu

ABSTRACT

Traditionally, graph centrality measures such as betweenness centrality are applied to discrete, static graphs, where binary edges represent the 'presence' or 'absence' of a relationship. However, when considering the evolution of networks over time, it is more natural to consider interactions at particular timesteps as observational *evidence* of the latent (i.e., hidden) relationships among entities. In this formulation, there is inherent *uncertainty* about the strength of the underlying relationships and/or whether they are still active at a particular point in time. For example, if we observe an email communication between two people at time t, that indicates they have an active relationship at t, but at time $t + k$ we are less certain the relationship still holds. In this work, we develop a framework to capture this uncertainty, centered around the notion of *probabilistic paths*. In order to model the effect of relationship uncertainty on network connectivity and its change over time, we formulate a measure of centrality based on most *probable* paths of communication, rather than shortest paths. In addition to the notion of the relationship strength, we also incorporate uncertainty with regard to the transmission of information using a binomial prior. We show that shortest paths in a unweighted, discrete graph can be formulated using probabilistic paths with a prior and we develop an algorithm to compute the most likely paths in $O\left(|V||E| + |V|^2 \log |V|\right)$. We demonstrate the effectiveness of our approach by computing probabilistic betweenness centrality over time in the the Enron email dataset.

General Terms

Algorithms, Measurement

Keywords

Probabilistic Graphs, Time-Varying Graphs, Betweenness Centrality

The 4th SNA-KDD Workshop '10 (SNA-KDD'10) July 25, 2010, Washington D.C. USA

1. INTRODUCTION

An important concept in the study of networks and graphs is to identify the most important (i.e., *central*) nodes in the network. These central nodes are believed to facilitate the flow of information through a network, and finding them can have implications in fields ranging from viral marketing to modeling traffic in a city.

To date, much of social network research has focused on modeling discrete graphs, where edges represent the 'presence' or 'absence' of relationships. In many cases this is a reasonable choice of representation: a road between two cities exists, a power line connects a house to a power plant, and hyperlinks connect pages on the Web. However in other networks, for example time evolving social and communication networks, this discrete notion of edges may no longer be appropriate. In these types of networks, relationships can change over time—for example, strangers become friends while close friends drift apart.

When analyzing evolving networks, an obvious approach is to compute centrality by applying traditional (static) measures to the *aggregate* network, where edges accumulate over time (i.e., edge (i, j) is included if there has *ever* been a link between i and j). To compute betweenness centrality for a particular node the history of all edges is used to calculate the number of shortest paths in the graph that traverse the selected node. However, when there is uncertainty about the presence or 'activity' of a relationship at a particular point in time, it is important to measure the change in centrality over time, rather than basing measures of connectivity on links that are weak, outdated, or no longer active. Recent research on temporal centrality measures [14, 8] has used discretized timesteps to represent evolving networks, by dividing time into segments where a message occurring during a segment is observed and others are ignored. However, this approach can be problematic due to the inherent discrepancies and irregularity of messages (e.g., [8] introduces a 'jitter' parameter solely to overcome windowing effects).

In this work, we contend that *probabilistic* graphs are a more natural representation for temporally-evolving communication graphs. With this type of representation, we can model the *probability* of a friendship being active, based on the observed history of communication between the two nodes. Probabilistic graphs can also represent the latent (i.e., hidden) relationship *strength* among nodes, which has been investigated recently [15, 5]. This work has focused on predicting tie strength in online social networks using the characteristics of users and the history of interactions between them, resulting in estimated relationship *weights* that

vary between 0 (i.e., 'absent') and 1 (i.e., 'present'). Despite the inherent uncertainty in relationship presence, activity, and/or strength that is natural in many social network domains, there is no general framework to incorporate this uncertainty into network analysis tools. Here, we develop a notion of probabilistic paths in uncertain networks, and use it as a foundation for computing probabilistic betweenness centrality in networks evolving over time.

Initially, we compute probabilistic betweenness centralities by sampling graphs with estimated relationship strengths for a particular time t, then average the betweenness centrality rankings over the set of samples. We show that it takes relatively few samples to calculate a reasonable estimate of betweenness centrality.

Next, we develop an exact calculation of betweenness centrality for probabilistic networks, based on the notion of most *probable* paths. To motivate this approach, we note that conventional betweenness centrality uses shortest paths as an indication of how quickly information can potentially flow in the network. When adopting a probabilistic view of the network, information flowing across paths with *fewer* nodes is less important than whether the information is successfully *transmitted*. In this case, central nodes should correspond to nodes that have high probability of transferring information throughout the graph, regardless of path length. To encode this notion, we consider *probabilistic paths* and develop an algorithm for finding the *most probable* paths in our network. We note that our formulation, which models the probability of the spread of information across the graph, is consistent with the finding in [10], which identified that constricting and relaxing the flow along the edges in the network was necessary to model the true patterns of information in an evolving communication graph.

In addition, we note that current centrality measures, which focus on shortest paths, implicitly assume that a node is guaranteed to pass all available information to its neighbors [14, 7]. Intuitively, this assumption is unlikely to hold in social networks; people rarely update friends and coworkers about every aspect of their personal life. A similar argument applies to disease transmission in networks, where models of diffusion incorporate the likelihood of *contagion* into calculations of flow and spread. Consequently, the likelihood of *transmission* should be incorporated into our calculations of node centrality. To address this, we formulate a prior that accounts for the uncertainty associated with transmission of information along edges, and incorporate it into our centrality rankings. We note how the notion of transmission uncertainty along probable paths incorporates the naturally desirable qualities of (certain) shortest paths—longer paths are typically less likely to deliver information across the graph. We show that on discrete graphs our formulation of handicapping longer probabilistic paths with a prior is equivalent to the shortest path formulation commonly used.

Our proposed centrality methods have fairly good runtimes – the sampling formulation takes $O\left(m \cdot |V|\,|E|\right)$, where m is the number of samples taken, and we show that relatively few samples are needed to obtain a good estimate. Both most probable paths and most probable handicapped paths can be computed in $O\left(|V|\,|E| + |V|^2\,|E| \log |V|\right)$, with modified versions of Dijkstra's algorithm [3] for most probable paths, and Brandes' algorithm [1] for betweenness centrality.[1]

Within the proposed framework, we investigate the centrality of individuals over time in the Enron email dataset. Our analysis shows that our probabilistic formulation offers the following advantages:

- Smoother transitions in centrality ranking, when compared to centrality calculated on discretized time slices.
- More accurate characterization of temporal centrality, when compared to centrality calculated on the aggregate graph.

The remainder of the paper is as follows: related work is described in Section 2, probabilistic graphs and paths are in Section 3. Section 4 outlines the sampling method, as well as the notion of most probable paths. In Section 5 we analyze the Enron email dataset, and in Section 6 we conclude.

2. RELATED WORK

Determining the centralities of nodes in networks has been extensively studied, with metrics ranging from simple rankings based on the degree of the nodes, to more complicated methods which involve computing eigenvectors [2]. Despite the extensive research into centrality measures, relatively little has been done in finding the centrality of *time-evolving* networks, with the notable exceptions being [14] and [7]. [14] formulates the problem of finding the most central nodes *throughout* time, by using the notion of a temporal graph. To create this temporal graph, the edges are collapsed by day, and shortest paths are found through the days. However, this makes an assumption that the most important nodes in a network remain the most important throughout time; a notion which is unlikely to hold as we are able to examine networks as they evolve throughout more extensive time periods. In fact, in our analysis we can find specific people who are *only* important with key events in time, rather than being the most important throughout time. The vector clock method proposed by [7] formulates this notion by developing a temporal notion of important edges based on an edge's ability to convey information directly between nodes faster than bypassing along alternate paths. However, this does not lead us to be able to determine which set of nodes and edges are the most 'central' in the network.

When developing probabilistic paths, we note that the notion of probabilistic graphs have been studied previously, notably by [4], [6] and [11]. [4] showed how when we have graphs with probability distributions of weights for each edge, we can use Monte Carlo to sample to determine the shortest path probabilities between the edges. [6] then extends on this to find the paths most probable to complete within a certain time constraint. Determining the most probable shortest paths when the edges have a probability of *existence* is closely related to determining the most probable shortest paths when we have a *distribution* of weights, and Monte Carlo can be used for both [11, 4]. Interestingly, [11] chooses to weight the shortest path distribution based on the probability of the sampled graph's existence, rather than keeping in line with typical sampling techniques. We choose to follow the formulation by [4], where every sampled graph gets equal weight. We note that our formulation for

[1]This is the same complexity as computing betweenness centrality on positively weighted graphs [1].

most likely probable paths takes advantage of the fact that our graphs are *unweighted*; any network which has weights and probabilities would need a sampling approach.

3. PROBABILISTIC GRAPHS

Generally, we define a graph $G = \langle V, E \rangle$, where V is a collection of nodes and E is our collection of edges, or relationships, between the nodes. Many times, weights are assigned along the edges, so an edge e_{ij} which connects node v_i and v_j would be assigned a weight $w(e_{ij})$. In social networks, we rarely have weights assigned to an edge, rather, people communicate with each other through messages, which we can use to indicate the probability of an active relationship between the two nodes [15]. We see that we now have two separate and distinct notions; *edges* and *messages*. We define an edge e_{ij} to be the probabilistic connection between two nodes, indicating whether the nodes have an active relationship. This is in contrast to messages; a message m_{ij} is an concrete and directly measurable communication between two nodes v_i to v_j. An edge is inherently unobservable, we can estimate the probability of an active relationship between two nodes, but this has to be inferred from the characteristics of the nodes and messages sent between them.

Furthermore, both have different interpretations with respect to time. Messages between nodes occur at a specific time, which we denote as $t\left(m_{ij}^k\right)$. A message can *indicate* an active relationship at the current time $t(\mathbf{now})$, but it itself does not *occur* at $t(\mathbf{now})$. This is in contrast with edges, the relationship strength indicated by an edge can be asked for at *any* time; it is not fixed. We denote this to be $e_{ij}^{t(\mathbf{now})}$, and we frequently simplify this to be e_{ij}^t. We can then denote $P\left(e_{ij}^t\right)$ to be the probability of an edge at a specific time. Probabilistic paths are not dependent on using time, so we can refer to this as $P(e_{ij})$ when time is not a necessary factor.

3.1 Probabilistic Paths

A path is defined to be a sequence of vertices such that from each vertex to the next there exists an edge; we define $V(\rho_{ij})$ and $E(\rho_{ij})$ to be the vertices and edges that constitute a path ρ_{ij}, which denotes the path between two nodes v_i and v_j. Extending paths to the probabilistic framework, we make the assumption that the probability of an edge existing is independent of all other edges. We can now define the probability of a path over a probabilistic graph to be the multiplication of the probabilities of the edges along the path. Formally,

$$P(\rho_{ij}) = \prod_{e_{k,k+1} \in E\left(\rho_{ij}\right)} P(e_{k,k+1}) \tag{1}$$

It is apparent that this notation applies to discrete paths as well as probabilistic paths, which is discussed in detail in Section 4.4. We can see that if we were to able to calculate the distribution of graphs defined by the probabilities on the edges, ρ_{ij} would exist with probability $P(\rho_{ij})$; the probability that all edges exist simultaneously.

4. PROBABILISTIC GRAPH CENTRALITY

As with discrete graphs, frequently we are asked to identify the nodes which are central to facilitating information flow through the graph, and in a time evolving graph, we would like a way to compute the most central nodes at a specific juncture as well. It is clear that the usage of the probabilities as weights is incorrect, as [11] notes. Instead, we create two alternate methods based around the notion of probabilistic paths: sampling and most probable paths.

4.1 Sampling for Probabilistic Centrality

Let $\mathcal{G}$ be the given probabilistic graph, which defines a distribution of discrete, unweighted graphs. Each unweighted graph $G \in \mathcal{G}$ has a betweenness centrality ranking for every node v_i, which we denote $\mathrm{BCR}_i(G)$. Next, $\mathrm{BCR}_i(\mathcal{G})$ is defined as being a random variable for the betweenness centrality of a node v_i over the distribution of graphs $\mathcal{G}$. The expectation for the random variable $\mathrm{BCR}_i(\mathcal{G})$ is given by:

$$\mathbb{E}\left[\mathrm{BCR}_i(\mathcal{G})\right] = \sum_{G \in \mathcal{G}} \mathrm{BCR}_i(G) \cdot P(G) \tag{2}$$

Typically, the distribution of the graphs $G \in \mathcal{G}$ is intractable to compute directly, so we sample to approximate our expectation. Given that we draw m sample graphs $\hat{G}$ independently from the distribution $\mathcal{G}$, we get a uniform probability for the sample graphs. We can now approximate the expectation for the betweenness centralities:

$$\mathbb{E}\left[\mathrm{BCR}_i(\mathcal{G})\right] \simeq \frac{1}{m} \sum_{\hat{G} \in \mathtt{G}} \mathrm{BCR}_i\left(\hat{G}\right)$$

Due to the fact that each of our sampled graphs are unweighted, the computation of the betweenness centrality for each is $O(|V||E|)$, for an overall cost of $O(m \cdot |V||E|)$. This is more expensive than typical sampling costs; however, we need to draw relatively few samples to get a good estimate of the betweenness centrality (section 4). Finally, we note that this sampling method is applicable for networks that have weights, for the cost of $O\left(m \cdot \left(|V||E| + |V|^2 \log |V|\right)\right)$.

4.2 Most Probable Path

Using the definition of probabilistic paths, we are now interested in the notion of the *most probable* path. That is, if vertex v_i sends out a piece of information, what path is the *most likely* to deliver the information to vertex v_j? If we have accurate estimates of relationship strength, and assume that all known information is transmitted from a node to its neighbors whenever it sees them, it is clear that the most probable path tells us which path is the most probable for delivering the information from v_i to v_j. It is important to note that, to our knowledge, all previous research in temporal paths also makes this implicit assumption of transmitting the information perfectly from one node to the next [7, 14].

A key advantage to using the most probable path formulation described is that we can precisely calculate *all* of the most probable paths, between every vertex, in exactly $O\left(|V||E| + |V|^2 \log |V|\right)$. Algorithm 1 outlines how to do this; we modify Dijkstra's shortest path algorithm [3] by selecting the most likely unvisited path, rather than the shortest unvisited path. Additionally, we can modify Brandes' algorithm [1] to start with the path that has the lowest probability of occurrence to be the one to backtrack from, allowing for computation of the betweenness centrality in $O\left(|V||E| + |V|^2 \log |V|\right)$ as well.

Algorithm 1 ML_Paths

Input: Index for some node i
Output: Probability of ρ_{ij} for all v_j
Backpointers for recreation of ML Paths

array $path\_probs = [0, \ldots, 0]$
array $visited = [false, \ldots, false]$
array $previous = [-1, \ldots, -1]$
$path\_probs\,[i] = 1$
while there are unvisited nodes **do**
  Set cur to max in $path\_probs$ and be unvisited
  for all o who are unvisited neighbors of cur, **do**
    if $path\_probs\,[cur] \cdot e_{cur,o} > path\_probs\,[o]$ **then**
      // Update the most probable path
      $path\_probs\,[o] = path\_probs\,[cur] \cdot e_{cur,o}$
      $previous\,[o] = cur$
    end if
  end for
  $visited\,[cur] = false$
end while
return $previous$, $path\_probs$

Algorithm 2 ML_Handicapped_Paths

Input: Index for some node i
Output: Probability of ρ_{ij} for all v_j
Backpointers for recreation of ML Paths

array $path\_probs = [0, \ldots, 0]$
array $posterior\_probs = [0, \ldots, 0]$
array $path\_length = [0, \ldots, 0]$
array $visited = [false, \ldots, false]$
array $previous = [-1, \ldots, -1]$
$path\_probs\,[i] = 1$
while there are unvisited nodes **do**
  Set cur to max in $posterior\_probs$ and be unvisited
  for all o who are unvisited neighbors of cur, **do**
    $p\_prob = path\_probs\,[cur] \cdot e_{cur,o}$
    $p\_length = path\_length\,[cur] + 1$
    $p\_post = p\_prob \cdot \mathrm{SBin}\,(p\_length|\beta)$
    if $p\_post > posterior\_probs\,[o]$ **then**
      // Update the most probable path
      $path\_probs\,[o] = p\_prob$
      $path\_length\,[o] = p\_length$
      $posterior\_probs\,[o] = p\_post$
      $previous\,[o] = cur$
    end if
  end for
  $visited\,[cur] = false$
end while
return $previous$, $path\_probs$

4.3 Transmission Uncertainty in Longer Paths

As stated previously, both the most probable path formulation and other temporal formulations of centrality, make the implicit assumption that *all* information is transmitted across an edge. However, this may be a poor assumption, since in a social network, people rarely transfer all information about themselves to their contacts. Consider the case where there is a chain of 10 people all with high relationship strengths, and a different path of length 2 where the relationship strength between the people involved is moderate. We would expect that information is more likely to flow along the shorter path, solely because it is unlikely that a message would make it through the entire length of the longer path. What needs to be accounted for is the probability of transmission, which incorporates the probability of a person conveying information to a friend into the path probability. This gives us additional insight into why shortest paths are important; there is higher likelihood that the information will be transmitted if it goes through fewer people.

The Binomial Distribution . For every step in a particular path, we can assign a probability β of success; or a probability that information is transmitted across an edge and is received by the neighboring node. If we denote l to be the number of steps in our path, and s to be the number of successful transmissions along the path, our formulation for the binomial distribution becomes:

$$\mathrm{Bin}\,(s|l,\beta) = \binom{l}{s}\beta^s\,(1-\beta)^{l-s} \tag{3}$$

$$\text{where: } 0 < \beta < 1$$

We see that when viewing probabilistic paths, we are only concerned with the single case where it *always* succeeds. As such, our prior simplifies to be:

$$\mathrm{SBin}\,(s|\beta) = \beta^s$$

Using the binomial distribution models our expected probability of information spread in an intuitive way, giving us a parameter β which we can adjust to fit our expectations for the information spread in the graph. Note that setting $\beta = 1$ leaves us with the original probabilistic paths notation, illustrating how the binomial is an extension of our original formulation.

In addition to the advantage of incorporating transmission uncertainty into our paths, we note that the prior has the *effect* of handicapping longer paths through the graph. We can find correlation between shortest (certain) paths and handicapped (uncertain) paths; however, we note that these are *not* the same formulations. We believe that it is a natural cross between the discrete graph and the probabilistic graphs; in the discrete graph shortest paths implicitly decrease uncertainty, just as our explicit modeling of transmission uncertainty does here.

ML Handicapped Paths. Now that we have both the notions of a probabilistic path, and an appropriate prior for modeling the probability of information spreading along the edges in the path, we can formulate the maximum likelihood handicapped path between two nodes v_i and v_j to be:

$$\mathrm{ML}\,(v_i, v_j) = \max_{\rho_{ij}}\left[P\,(\rho_{ij}) \cdot \mathrm{SBin}\,(L\,(\rho_{ij})\,|\beta)\right] \tag{4}$$

where $L\,(\rho_{ij})$ is the length of the path. In Algorithm 2 we give the formulation for this, which requires keeping track of the probability of a path separately from the posterior. Like the most likely paths formulation, we can calculate both the most likely handicapped paths and the corresponding betweenness centrality in $O\left(|V|\,|E| + |V|^2 \log |V|\right)$.

4.4 Handicapped Paths and Shortest Paths

The formulation of ML Handicapped Paths has inherent

benefits, most notably with its direct connection to the previously well-studied notions of shortest paths and betweenness centrality. In fact, we can view the discrete graph as being a specific case of the probabilistic graph. To do this, we define the probabilities for discrete edges to be:

$$P(e_{ij}) = \begin{cases} 1 & \text{if an edge exists} \\ 0 & \text{if the edge does not exist} \end{cases} \quad (5)$$

This is an intuitive extension; if an edge exists in a discrete graph, than we know with probability 1 that it is there. Likewise, if an edge is not present, than it almost surely will never exist.

THEOREM 1. *Equivalent Paths*
The set of probabilistic paths $\widetilde{\mathcal{P}} = \{\rho | P(\rho) = 1\}$ is precisely the same as the static paths $\mathcal{P}$, on a discrete graph with probabilities defined by Equation 5.

PROOF. The base case is defined to be $L(\rho) = 1$. Since $\forall\, e \in E:\ P(e) = 1$, these must also be in $\widetilde{\mathcal{P}}$. For all edges e' not present in E, $P(e') = 0$, so if an edge is not present in E, than it is also not present in $\widetilde{\mathcal{P}}$.

Set ρ_i to be the first path where $\rho_i \in \mathcal{P} \wedge \rho_i \notin \widetilde{\mathcal{P}}$. Since ρ_i is the first false case, we know $\rho_{i-1} \in \mathcal{P} \wedge \rho_{i-1} \in \widetilde{\mathcal{P}}$. $e_{i-1,i}$ must be present in order for $\rho_i \in \mathcal{P}$. The probability of $e_{i-1,i}$ was defined to be 1, so $P(\rho_{i-1}) \cdot P(e_{i-1,i}) = 1$, so $\rho_i \in \widetilde{P}$, a contradiction. So the proposition holds.

The opposite case, where $\rho_i \in \widetilde{\mathcal{P}} \wedge \rho_i \notin \mathcal{P}$ can be proved using a similar proof. □

THEOREM 2. *Shortest Paths and Handicapped Paths*
The shortest path in the static graph for $\rho_{ij} \in \mathcal{P}$ is the same as the ML Handicapped Paths formulation, when we choose $0 < \beta < 1$.

PROOF. Using Proposition 1, we know that the paths in $\mathcal{P}$ from v_i to v_j are precisely the same as those found in $\widetilde{\mathcal{P}}$ where $P(\rho_{ij}) > 0$. Furthermore, because every $P(e_{ij})$ is either 1 or 0, every case where $P(\rho_{ij}) > 0$ is precisely $P(\rho_{ij}) = 1$. For the definition of shortest path in $\mathcal{P}$, we know that we have minimized $L(\rho_{ij})$. Since the probability of a path is always going to be 1 in this setting, the only thing that can affect the posterior is the prior. Our prior is defined in (4), with the assumption that $0 < \beta < 1$. Since we are multiplying β additional times for any path that is longer than the shortest path, our posterior must be lower for any path that is longer than the shortest path. As such, both formulations produce the same shortest path. □

THEOREM 3. *Equivalent Betweenness Centrality*
The betweenness centrality using $\mathcal{P}$ can be equivalently calculated with $\widetilde{\mathcal{P}}$, where edge probabilities on the discrete graph are defined by Equation 5

.

PROOF. This follows directly from proposition 2; since we have the same shortest paths, and they all have $P(\rho_{ij}) = 1$. □

In addition to the proofs for the connections between ML Handicapped paths and shortest paths, we show the correctness and time complexity for the ML Handicapped paths algorithm.

THEOREM 4. *ML Paths Correctness*
The ML Paths algorithm and ML Handicapped Paths algorithm finds the maximum likelihood paths.

PROOF. We note the step in Algorithm 1 where we choose the maximum probability of unvisited nodes; say ρ_{ij}. If we were to choose another path which had a lower probability (for instance, ρ_{ik}), our formulation becomes $\rho'_{ij} = \rho_{ik} \cdot \rho_{kj}$. Clearly, the maximum probability that ρ_{kj} can have is 1, so the maximum of $\rho'_{ij} = \rho_{ik}$. However, we know that $\rho_{ik} < \rho_{ij}$; therefore, $\rho'_{ij} < \rho_{ij}$ and is not optimal. This holds as well for ML Handicapped Paths, where we now have an additional β penalizer for the additional step. □

THEOREM 5. *ML Paths Time Complexity*
The ML Paths algorithm and ML Handicapped Paths can be solved in $O\left(|V|\,|E| + |V|^2 \log |V|\right)$ for all ρ_{ij}.

PROOF. [1] showed that we can compute betweenness centrality in for weighted graphs in $O\left(|V|\,|E| + |V|^2 \log |V|\right)$. Here, we choose the most probable paths, rather than the shortest paths. As such, the computation cost does not change. □

5. EXPERIMENTS

For our analysis we use the Enron dataset compiled by Shetty and Adibi [13]. In addition to the fact that this dataset is time-evolving, it allows us to study the effects of our centrality measures unlike other datasets, in the sense that key events and central people have been well documented [9]. The dataset itself was originally posted by the Federal Energy Regulatory Commission during its investigation of Enron, and contained upwards of 800,000 emails among 151 ex-employees of Enron; many of the emails have since been deleted at the request of employees. The current version contains 517,431 emails from 151 employees. We are only interested in the emails that were sent from employee-employee (or multiple employees), and exclude outside emails.

For comparison, we use four algorithms. First is the *aggregate* method, which at a particular time examines the entire graph that it has observed so far, or evaluating the standard betweenness centrality measure. Next, we take *slices* of time, where we only consider the messages that occurred within a selected time period before the time being evaluated. Additionally, we evaluate both the ML handicap and sampling methods defined earlier.

Time Slices. For the discretized time slices, we choose a period of 14 days to be in each time slice in our evaluation. Our intuition on why this is a good duration is that the slice will contain two full work weeks. Since periods of email flow in corporate settings are certainly dependent on what days of the week that meetings are held, at least a week is necessary, likewise, a month seems inherently too long to capture changes as they occur.

Relationship Strength. In order to evaluate both the sampling method and the most likely paths method, we need a measure of relationship strength for our model. Although any notion of relationship strength can be substituted at this step; we investigate the performance of the framework by defining a relatively simple relationship strength. First, define the exponential decay for a particular message to be:

$$\text{Exp}\left(m_{ij}|t(\textbf{now})\right) = \exp\left\{\frac{1}{\lambda}\left(t\left(\textbf{now}\right) - t\left(m_{ij}\right)\right)\right\}$$

This exponential decay for a single message indicates the probability of having an active relationship between nodes v_i and v_j at the current timestep, based on message m_{ij}. Formally, we get:

$$P\left(e_{ij}^{t}|m_{ij}\right) = \text{Exp}\left(m_{ij}|t(\textbf{now})\right)$$

$$P\left(e_{ij}^{\bar{t}}|m_{ij}\right) = \left(1 - \text{Exp}\left(m_{ij}|t\left(\textbf{now}\right)\right)\right)$$

where $P\left(e_{ij}^{t}|m_{ij}\right)$ indicates an active relationship at $t\left(\textbf{now}\right)$ given a message m_{ij}, and $P\left(e_{ij}^{\bar{t}}|m_{ij}\right)$ indicates the probability of *not* having an active relationship at $t\left(\textbf{now}\right)$. Next, rather than only considering a *single* message m_{ij}, we can consider *all* of the previous messages between nodes v_i and v_j in our formulations. Specifically, if *any* of the previous messages indicate an active relationship, than our edge would be considered to have an active relationship. In order to not have an active relationship, all of the previous messages would indicate not having an active relationship. Formally,

$$P\left(e_{ij}^{\bar{t}}|m_{ij}^{1},\ldots,m_{ij}^{k}\right) = \prod_{k}\left(1 - \text{Exp}\left(m_{ij}^{k}|t\left(\textbf{now}\right)\right)\right)$$

$$P\left(e_{ij}^{t}|m_{ij}^{1},\ldots,m_{ij}^{k}\right) = 1 - P\left(e_{ij}^{\bar{t}}|m_{ij}^{1},\ldots,m_{ij}^{k}\right)$$

where we have k messages occurring prior to time $t\left(\textbf{now}\right)$.

5.1 Parameter Setting

Setting λ. In order to set the scaling parameter λ for relationship strength, we took each employee and measured the correlation between the sampling method rankings and time slice rankings for that particular employee. The correlations between the rankings for all 151 employees were then averaged. An identical method was used to measure the correlation between the sampling method and the aggregate method, and the results are shown in Figure 1.a. We note that the left side of Figure 1.a indicates the situation where we have an extremely small scaling parameter λ; this is the situation where we 'forget' a message quickly. The right side similarly corresponds to the case where a message is given weight for long periods of time. As expected, small values of λ clearly shows high correlations with the 'slice' rankings, while larger values approach a correlation of 1 with the aggregate. Notice that as we get smaller and smaller λ, the correlation with the slice rankings *decreases*; this is the case where our λ is much smaller than our slice duration, and the slices can no longer keep track of the most immediate changes. We note that when $\lambda = 14$, identical to the time slice, it has much higher correlation with the aggregate than the slice method does at that time. In fact, we have to choose $\lambda \approx 3.5$ to reach the point where it has lower correlation with the aggregate than the slice method of 14 days, because the slice method has such high variability between slices. We want to be able to balance between short term change and long term trends by setting λ to a 'middle ground'. We note that we achieve this balance around $\lambda = 28$, where the two cross.

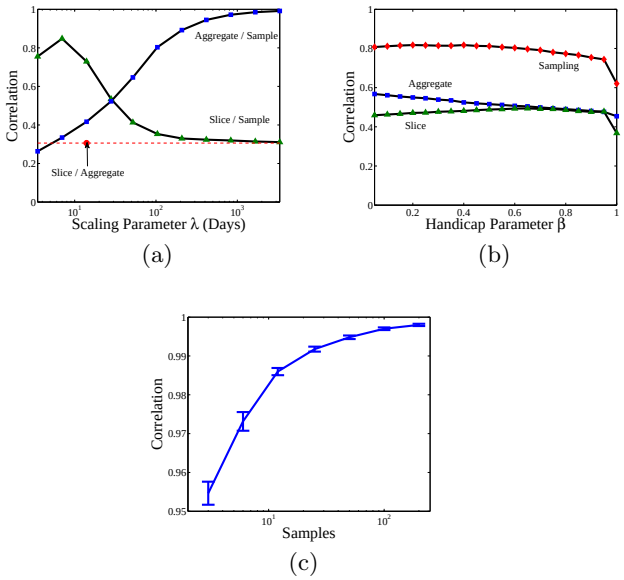

Figure 1: Correlation between the Sampling Method and the Aggregate/Slice Methods for varying values of λ ((a) and (b)). Number of samples and correlation with the 10,000 samples (c).

Number of samples. First, we note that [11] found that relatively few samples are needed to compute a decent estimate of the shortest paths in the graph, based on the Hoeffding Inequality. We note that our betweenness centrality rankings are based on shortest paths, so they also need relatively few samples in order to get a good estimate. We show this empirically, using the Enron Dataset (discussed in section 4) which contains 151 nodes. Figure 1c demonstrates the relative accuracy of the rankings; we see that with just a handful of samples, the rankings are highly correlated with the rankings produced by taking 10,000 samples. Given that some nodes have no change in betweenness centrality, this overestimates the correctness to some extent, but is a reasonable measure. As expected, more samples gives a better estimation, so for setting λ, we used 200 samples, and for further analysis we used the full 10,000 sample set.

Setting β. Here, we note the need for choosing a transmission parameter β, for use with the ML Handicapped Paths (MLH) algorithm. For this, we have already set $\lambda = 28$ for both the sampling and the MLH algorithms; the correlations between the MLH algorithm and the alternative algorithms are shown in Figure 1.a. As expected, no matter what value we choose for β our correlation is higher with the sampling method than either the aggregate or slice methods. This is partly due to our choice of λ; we also found that a smaller value of λ resulted in approximately even correlation between the slice and MLH methods against the sampling method. Likewise, a large λ results in high correlation between aggregate/sampling/MLH methods. Both of the probabilistic approaches allow for balance between short-term change and long-term information. For our subsequent evaluations, we set $\beta = .3$, which is approximately where peak correlation happened.

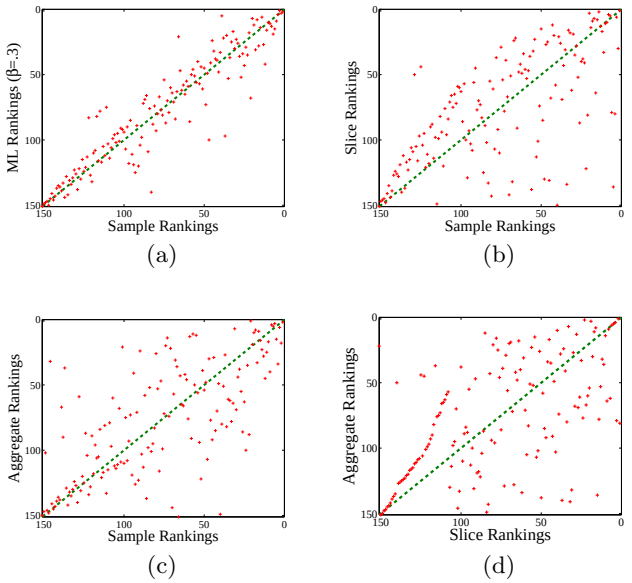

Figure 2: Correlations of the Rankings for the time segment ending at August 24^{th}, 2001

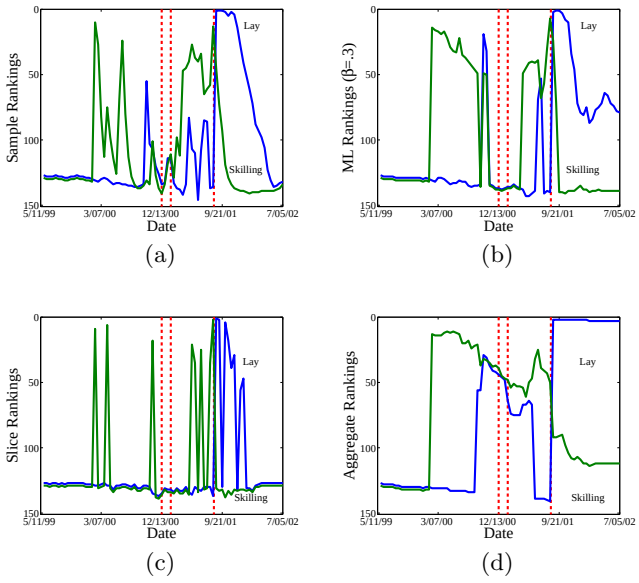

Figure 3: Time rankings of Lay and Skilling. Red lines indicate Skilling's CEO announcement, takeover and resignation.

Correlations on August 14^{th}, 2001. In order to illustrate the differences between the four methods, we analyze their respective rankings at a particular point in time: August 14^{th}, 2001. This was a particularly tumultuous time for Enron (see below in Section 5.2) and we can use it to see how closely the rankings between methods match (shown in Figure 2). The x-axis indicates the rankings for one of the methods, from lowest rankings (left) to highest (right), while the y-axis indicates the rankings for a different method, again from lowest rankings (bottom) to highest (top). The green line from bottom-left to top-right indicates the 'perfect' correlation; if the two measures are identical, all of the red +'s will lie directly on top of it, the farther we stray from the green line the less correlated the methods.

We can see in Figure 2.a that the MLH method closely approximates the sampling method, with only a few people's rankings varying from the diagonal, while Figure 2.b indicates that the slice method only somewhat approximates it. We can see that the slice method follows the sample method to some extent; however, a large number of nodes with high centrality from the sampling method are missed by the slice method. This highlights the problems with using the slice method; nodes with high centralities are liable to be overlooked due to the imprecision of discretized time. Additionally, we note that August 14^{th}, 2001 is relatively late in our dataset. The aggregate method has little correlation with the slice method and the sampling method, giving weight to our intuition that the aggregate is incapable of tracking *current* events in the network.

5.2 Lay and Skilling

As stated previously, the Enron dataset is unique and useful for us in the sense that we know the background of the timeline, and can therefore investigate the correctness of centrality measures. In this case, we choose to analyze two key figures at Enron: Kenneth Lay and Jeffery Skilling. Both were key figures in the Enron scandal and, more importantly, Lay and Skilling *traded off* being CEO of the company: Lay was CEO first, then handed it off to Skilling. Several months later, Skilling resigned as CEO, relinquishing control back to Lay. We can analyze the centrality rankings for Lay and Skilling during these transition periods, as we expect large changes to happen to both of them. We note that during routine moments in time, Lay has a relatively low centrality, indicating that a secretary or someone is handling generic communication with other people. However, during the eventful transition periods, Lay has much higher centrality, indicating his desire to communicate with other executives directly.

December 13^{th}, 2000. On December 13^{th} it was announced that Skilling would assume the CEO position at Enron, with Lay retiring but remaining as a chairman [9]. We note that in Figure 3.a, both the sampling method and the handicap method identify a spike in activity for both Lay and Skilling directly before the announcement. This is not surprising; presumably Skilling and Lay were informing the other executives about the transition that was about to be announced. Additionally, after the announcement the sampling and handicap measures have Skilling and Lay's centralities decrease. This is due to both the holiday season and Skilling's trip to Houston [9], where he discussed with analysts the 'outstanding' fiscal shape of Enron.

We note that the time slice method makes no change in Lay's activity, despite his central role in the transition (3.c). Additionally, Skilling is given a few random spikes of activity, showcasing the unevenness of using the time slices. The aggregate model seems to do a good job picking up Lay's increased activity, but fails to reduce his activity to the expected levels following the announcement (3.d). This is fairly early in the temporal evolution and we are already

seeing the aggregate's inability to track current events. We also note the MLH is somewhat smoother than the sampling method, indicating the promise of using the most probable paths, rather than the probabilistic shortest paths.

February, 2001. During February, 2001, Skilling made the transition to CEO of Enron and Lay retired. We note following this date, Skilling's centrality increased. This can be attributed to either returning to his previous level of activity (after the Houston meetings), his promotion to CEO, or both. Additionally, Lay is now retired as CEO, so a fairly low centrality is not surprising. Both the sampling method and the handicap methods capture this; MLH has him return to an extremely low centrality, while sampling has him moving around somewhat, and is still somewhat rougher than MLH.

For this time period, the slice method hardly notices Skilling's position as CEO, and fails to give any notice to Lay, even though he is clearly a central figure in transition. Likewise, the aggregate has Skilling's centrality dropping during his CEO time, which is rather unlikely to occur.

August 14th, 2001. Seven months after initially taking the CEO position, Skilling approached Lay about resigning ([9]). During the entirety of Skilling's tenure, we see that Lay has a slight effect on the sample rankings, but is not what we would consider a 'central' node. Not surprisingly, Skilling has a fairly high centrality during his time as CEO; both the sampling method and handicap method capture this. After Skilling's resignation, note that *all* of the methods do manage to capture Lay's sudden increase in centrality.

Prior to the announcement of Lay's takeover as CEO, the slice method still had no weight on him, despite his previous involvement with the first transition. Additionally, we note that the sampling, handicap, and slice methods all agree that after Lay's initial spike from the Skilling resignation, he resumes having a lower centrality. The aggregate graph is simply unable to capture this change, and leaves Lay near the top of the centrality rankings.

5.3 Kitchen and Lavorato

Louise Kitchen and John Lavorato were executives [13] for Enron Americas, which was the wholesale trading section of Enron [12]. They are notable because of the extraordinarily high bonuses they received as Enron was being investigated, and were also found to have a high temporal betweenness centrality using the method defined by [14]. We can see in Figure 4 the rankings of Kitchen and Lavorato, and can key in on the benefit of using the probabilistic framework's ability to key in on centralities at *specific* times, rather than using the temporal definition *through* time proposed by [14]. We see that while Lavorato might have gotten a large bonus, he is *only* important during Skilling's tenure as CEO; his centrality drops noticeably otherwise. On the other hand, Kitchen had extremely high rankings throughout. This suggests that Skilling and Lavorato were extremely close, and Lavorato was therefore much more important when Skilling was running Enron. Our formulation of the probabilistic paths framework allows for this additional insight into the dynamic structure of the network. Once, we see that the probabilistic shortest paths is somewhat rougher than the most probable paths formulation,

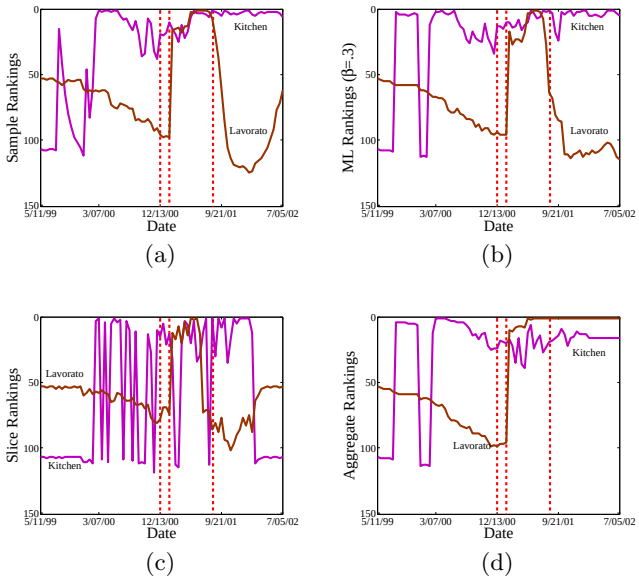

Figure 4: Time rankings of Kitchen and Lavorato.

specifically for Louise Kitchen.

Again, we see the inherent problems with using the slice or aggregate methods. The slice method is extremely finicky with regard to Kitchen, oscillating between an extremely high ranking and an extremely low ranking, while the aggregate method has Lavorato hitting high centrality at the correct time, but does not recognize the lack of importance of Lavorato after Skilling's departure.

6. CONCLUSIONS

In this paper we investigated the problem of calculating centrality in a time evolving network. We demonstrated the aggregate graph's inability to capture changes as they occurred, and the extreme variability of discretizing time. Our sampling approach - based on probabilistic graphs - allows for the ability to observe both the immediate changes as they occur, while incorporating smoothness through time. The sampling approach allows for the use of positive weights on our edges, making it extendable to applications beyond social networks.

In addition, we introduced the ML Paths and ML Handicapped Paths formulations, which have significantly higher correlation to the sampled graph than either the discretized time slices or the aggregate graph, and show additional smoothness not present in our sampling method. We proved that the shortest paths algorithm on unweighted, discrete graphs can be equated to be a specialized instance of the ML Handicapped Paths formulation.

We provided empirical evidence on the Enron dataset showing the sampling and MLH's intuitive centrality rankings for the Enron employees. Both the sampling and handicap formulations are inherently smoother than the discretized time slices, and allow for representing changes over time, unlike the aggregate method. We see the MLH formulation is smoother than the sampling method, indicating that the most probable paths through the graph are more important than the shortest. Finally, we note that our experiments

used a relatively simple estimate of relationship strength. Future work can be done on exploring the impact of different measures of relationship strength. a more elegant approach to model relationship strength at various timesteps.

Acknowledgements

This research is supported by DARPA and NSF under contract number(s) NBCH1080005 and IIS-0916686. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation hereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements either expressed or implied, of DARPA, NSF, or the U.S. Government. Pfeiffer is supported by a Purdue University Frederick N. Andrews Fellowship.

7. REFERENCES

[1] Brandes, U. A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology 25* (2001), 163–177.

[2] Brin, S., and Page, L. The anatomy of a large-scale hypertextual web search engine. In *Computer Networks and ISDN Systems* (1998), pp. 107–117.

[3] Dijkstra, E. W. A note on two problems in connexion with graphs. *Numerische Mathematik 1* (1959), 269–271.

[4] Frank, H. Shortest paths in probabilistic graphs. In *Operations Research, Vol. 17, No. 4 (Jul. - Aug., 1969), pp. 583-599* (1969), INFORMS.

[5] Gilbert, E., and Karahalios, K. Predicting tie strengthwith social media. In *CHI '09* (2009).

[6] Hua, M., and Pei, J. Probabilistic path queries in road networks: traffic uncertainty aware path selection. In *EDBT '10: Proceedings of the 13th International Conference on Extending Database Technology* (New York, NY, USA, 2010), ACM, pp. 347–358.

[7] Kossinets, G., Kleinberg, J., and Watts, D. The structure of information pathways in a social communication network. In *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining* (New York, NY, USA, 2008), ACM, pp. 435–443.

[8] Lahiri, M., and Berger-Wolf, T. Y. Mining periodic behavior in dynamic social networks. In *ICDM '08: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining* (Washington, DC, USA, 2008), IEEE Computer Society, pp. 373–382.

[9] Marks, R. Enron timeline. *http://www.agsm.edu.au/bobm/teaching/BE/Enron/timeline.html*.

[10] Onnela, J.-P., Saramäki, J., Hyvönen, J., Szabó, G., Lazer, D., Kaski, K., Kertész, J., and Barabási, A.-L. Structure and tie strengths in mobile communication networks. *Proc Natl Acad Sci U S A 104*, 18 (2007), 7332–6.

[11] Potamias, M., Bonchi, F., Gionis, A., and Kollios, G. Nearest-neighbor queries in probabilistic graphs, 2009.

[12] Raghavan, A., Kranhold, K., and Barrionuevo, A. Full speed ahead: How enron bosses created a culture of pushing limits — fastow and others challenged staff, badgered bankers; porsches, ferraris were big — a chart 'to intimidate people'. *http://academic.udayton.edu/lawrenceulrich/EnronBossesCreatingCulture.htm*.

[13] Shetty, J., and Adibi, J. The enron email dataset database schema and brief statistical report, 2004.

[14] Tang, J., Musolesi, M., Mascolo, C., Latora, V., and Nicosia, V. Analysing information flows and key mediators through temporal centrality metrics. In *Proceedings of the 3rd ACM Workshop on Social Network Systems (SNS'10)* (2010).

[15] Xiang, R., Neville, J., and Rogati, M. Modeling relationship strength in online social networks. In *Proceedings of the International World Wide Web Conference* (2010).