

Inferring Future Links in Large Scale Networks

Sima Das*, Sajal K. Das*, Susmita Ghosh†

* Department of Computer Science
Missouri University of Science and Technology
Rolla, Missouri 65409.
{sdp4b, sdas}@mst.edu

† Department of Computer Science and Engineering
Jadavpur University
Jadavpur, Kolkata, INDIA
susmitaghoshju@gmail.com; sghosh@cse.jdvu.ac.in

Abstract—The challenge in predicting future links over large scale networks (social networks) is not only maintaining accuracy, but also coping with the time-varying network graph. In contrast to the existing approaches, in this work we propose building a Markov prediction model. It not only incorporates temporal snapshots reflecting the dynamic network graph, but also considers effect of multiple timescales, along with corresponding local and global structural evolution (links and clusters respectively), correlated evolution and rate of evolution. The resulting edge selection in our approach exhibits the power law degree distribution, as exhibited in real world networks. Finally, we use two heavily dynamic real world network temporal data set (e.g. Twitter and Enron) and one relatively less dynamic network data set (e.g. DBLP), and existing state-of-the-art static and recent dynamic measures, to evaluate the prediction accuracy of our proposed Markov model and show that it out performs existing approaches.

I. INTRODUCTION

Large scale networks are inherently dynamic and ubiquitous in nature - ranging from biological networks, social networks, vehicular networks, and collaboration networks, etc. Underlying these application networks is the graph structure with genes or people etc. as nodes, and interaction between them as edges. The dynamic nature of these networks result from the addition/deletion of new nodes and edges over time, which is represented using time-varying graphs [1], [2], [3].

The importance of link prediction stems from its use in wide range of prediction applications: next collaboration between authors, new friendship in social networks, market trends and so on. Further, accurate link prediction facilitates efficient routing strategies in store-carry-forward networks minimizing delay and maximizing coverage area. Thus, creating massive research interest in automated, and accurate link prediction strategy [4], [5], [6], [7], [8], [9], [10].

Link prediction, though mostly addresses the possibility of new links between node pairs, it is also significant to address the recurring-links/ interactions over an existing link with examples like: interactions between two existing friends, collaboration between pair of earlier collaborators in an academic network. These time stamped recurring-links press the need for an incremental design model.

To set the stage for our novel contributions, we first present an overview of the existing literature on link prediction.

A. Related Work

Majority of the work in the literature of link prediction, consider static networks or static interpretation of a temporal

network that incorporate network structure in some way or other [4], [11], [12], [10], [6]. In these stationary models the basic concept is that the intrinsically dynamic network graph is mapped to an aggregated stationary graph over a time interval, in order to fit it to the stationary model. It ignores the time domain in individual interactions and their associated inter-relation. The network structure based preferential attachment model [13] is limited to connecting only pairs of high degree nodes with higher probability, and their network evolution model is based on this basic concept. Another approach considers Markov chains for link prediction [14]. The assumption of Markov property limits it from capturing structural evolution, and temporal dependency. The Markovian property considers only current state for prediction of next state, and limits the use of history states and underlying patterns in future prediction. Of the very few dynamic approaches, [15], [16] consider time series data to take into consideration the temporal evolution. Tylenda et al. [15], integrate temporal information with the local neighborhood based Adamic - Adar approach. Thus, in their graph based link prediction model they define the Adamic - Adar metric over temporal domain by taking into account the time stamps of interactions. They consider prediction of both new and recurring links. Whereas, in [16] the authors integrate time series with structural informations. The prediction model predicts links based on the features of its endpoints, and the local neighborhood around the end points.

The existing approaches are limited by the following key factors that motivate our approach.

B. Motivation

Our proposed model and associated mathematical formulae in Section III are inspired by the evolution of real-world large scale networks, with new and recurring links (interactions) over Facebook, Twitter, E-mail or academic collaboration networks etc.

In addition, most of the existing research consider aggregated representation of the underlying time-varying network to fit to the assumed stationary model [4], [11], [12], [10], [6]. In lieu of time-varying nature of large scale networks in the input graph, the prediction algorithm lacks its suitability to reflect the temporal evolution of the real world networks. Thus, leading to poor prediction accuracy over very dynamic networks (say mobile networks, opportunistic networks, Twitter or E-mail network).

Further, the existing dynamic approaches [15], [16] lack proper definition of time interval, thus the proposed prediction models lack the integration of time-varying property of the network with time domain. For example, evolution of a set of links over a time period leads to the evolution of clusters (a group of links with greater affinity) over that period, though evolution/ occurrence of individual links occur at distinct times. The conventional link prediction addresses whether two specific nodes indeed interact at a specific time, in contrast the generative models describe evolution on large scale using global characteristics (and latent parameters) without regard to which node interacts with which one. Among the graph based approaches, the prediction models are either network structure based [15] or are generative model based [17], [18], but what we really need is integration of both, for the most accurate and elaborate prediction model in determining the set of links at any future time. Further, the dynamic models [15], [16] restrict themselves to the effect of only local neighborhood on link prediction. The challenge is to incorporate the temporally evolving local and global structure (as the evolution of links and clusters over w.r.t. different time scales) in link prediction for improved prediction accuracy.

In this paper, we have the following contributions to address the above issues.

C. Summary of Contributions

First, we propose a probabilistic model, with time-varying graph as its basis, for both recurring and new link prediction over dynamic networks. Here, the link prediction probability is the transition probabilities in the corresponding Markovian transition matrix. We consider the state space of our Markov model to reflect the past s states during the evolution process and the model is built bottom up.

Second, for most accurate prediction model, we propose integration of different timescales over network evolution. Thus, the analysis considers both local and global network structure evolution with corresponding two-way temporal dimension; that is local structural evolution (link evolution) is analyzed over microscopic time domain, whereas global structural evolution (evolution of clusters) is analyzed over macroscopic time domain. Since, evolution of links gives rise to evolution of underlying communities/clusters, we have an intrinsically dynamic community structure. Besides, we also consider temporally correlated evolution, rate of evolution over the dynamic (time-varying) network for model construction. Thus, our model is able to capture very distinct evolution profiles during the observation period.

Third, we predict future links (both new and recurring links) over both microscopic and macroscopic time domain and take care of possible over-fitting. Further, we show that the predicted links exhibits power law degree distribution, as exists in real world networks.

Finally, we experimentally evaluate the prediction accuracy of our approach using three real world data sets with two more dynamic networks e.g. Twitter, Enron and a lesser dynamic data set, such as DBLP, and compare the accuracy with

existing static state-of-the-art and recent dynamic measures. Real data sets help us evaluate how well our model captures the underlying system dynamics to predict subsequent interactions in them.

The rest of the paper is organized as follows. In Section II, we formally define our problem. Section III describes our proposed Markovian model, the use of multiple time scales along with, correlated structural evolution and rate of evolution in link prediction, while Section IV presents the proof that our edge selections approach leads to power-law degree distribution. Section V reports experimental results and finally conclusions are offered in Section VI.

To the best of our knowledge, ours is the first work to consider Markovian matrix model that consider time-varying graphs with integration of microscopic and macroscopic temporal analysis for best accuracy. Further, we incorporate corresponding local and global structural evolution, correlated evolution, and rate of evolution to the prediction procedure.

II. PROBLEM DEFINITION

We represent the underlying network graph as $G(V, E, C, [0, T + 1])$, where V, E and C represent the set of vertices, edges and clusters respectively, $[0, T]$ is the training period and $[T, T + 1]$ is the test interval. At the beginning, in absence of any edge, $|C| = |V|$. In a temporal network, the network interactions are viewed as a set of discretized events over time axis. Let the interval $[0, T + 1]$ be divided into macroscopic time interval, called time slots $T_1, T_2, \dots, T_i, T_{i+1}, \dots, T_k$. Each of the time slot T_i is further divided into microscopic time interval, called time stamps $t_{1'}, t_{2'}, \dots, t_{i'}, t_{i'+1}, \dots, t_{k'}$, where time stamp reflects the smallest interaction time unit (say minute, hour, day or month) considered for an application. Let the corresponding subgraph over any time T_i and $t_{i'}$ be represented as $G_{T_i}(V_{T_i}, E_{T_i}, C_{T_i})$ and $G_{t_{i'}}(V_{t_{i'}}, E_{t_{i'}}, C_{t_{i'}})$ respectively, such that $G_{T_i}, G_{t_{i'}} \subset G$ and $G_{t_{i'}} \subset G_{T_i}$; $V_{T_i}, E_{T_i}, C_{T_i}$ and $V_{t_{i'}}, E_{t_{i'}}, C_{t_{i'}}$ are the respective set of vertices, edges and clusters, with $C_{T_i}^j \cap C_{T_i}^k = \emptyset$. For notational simplicity we represent the clusters $C_{T_i}^j, C_{T_i}^k \in C_{T_i}$ as C_i^j and C_i^k respectively. We consider directed network graph, where any undirected application network is incorporated as having bidirectional edges between any pair of nodes. From here onwards, we use link or interaction or edge or contact interchangeably.

Going by the observation of the structural evolution in real world networks, we make the following assumptions.

Model Assumptions:

- Network evolution process is a sequence of states.
- The start state has interactions to reflect reality.
- Every subsequent state is based on the past states (called history states).

Formal Problem Formulation:

We formally define the link prediction problem as the mapping $f : (G_{T_i}, C_{T_i}, \forall i \in [0, T]) \rightarrow \{L_{i+1}\}$, where G_{T_i}, C_{T_i} are respectively the subgraphs and clusters over corresponding

time slots T_i s, and $\{L_{i+1}\}$ is the set of links in the subsequent time slot $T + 1$ (i.e. interval $[T, T + 1]$). As we consider two distinct temporal axis, we consider $T + 1$ as both subsequent time slot T_{k+1} and time stamp $t_{1'}$ in T_{k+1} .

We represent the notations used in this paper in Table I.

TABLE I: Notation Explanation

Notation	Description
s	Set of states of the evolution process that are kept track
$[0, T]$	Observation period
T_i	Time slots in $[0, T]$
$t_{i'}$	Time stamps in T_i
$t_{k'}$	Last time stamp over all $t_{i'}$ w.r.t. each T_i
$t_{1'}$	First time stamp over all $t_{i'}$ w.r.t. each T_i
d_i	Number links for node v_i
d_j	Number links for node v_j
$d_{in}(v)$	In-degree of node v at time stamp $t_{i'}$
$d_{out}(v)$	Out-degree of node v at time stamp $t_{i'}$
$d(v)$	Total degree of node v at time stamp $t_{i'}$
C_i	Set of clusters in time slot T_i
C_i^j	A cluster j , with $C_i^j \subset C_i$
$MAXd(v)$	Maximum degree of node v
$MIN(a, b)$	Returns minimum of the two parameters a and b

III. MARKOVIAN MODEL IN LINK PREDICTION

In this Section we describe the details of our proposed stochastic prediction model, the steps involved and how they work in tandem in computing prediction probabilities.

A. Model Definition and Initialization

This Subsection discusses high-level integration of the temporal network with our proposed model, our model architecture, and its initialization process.

1) *General Model Definition:* We visualize the network evolution as a process, as is assumed above, going by the intrinsic evolution process in real world networks. Here, the network is going through a sequence of states, represented as time-varying graphs over respective time domain i.e. over time slots or time stamps. We can interpret the evolution as a transition process from one network state to another. Over a given a set of states, the correct subsequent state will depend on the accuracy of the transition probabilities, which is dependent on the past states. From the graph theoretic interpretation of the state as the set of links, and the transition probability as the probabilities for each node pair over successive states, the resulting transition probability between states (i.e. for all node pairs) is a transition probability matrix.

We thus propose a Markov prediction model, the states of which correspond to network states and the inter-state transition probability is represented as the Markovian transition matrix. The elements of the Markovian transition matrix reflect the prediction probability for links over subsequent state. For most accurate prediction probability, the state of the Markov model encapsulates the past s states within it.

We consider both local and global structural evolution (links and clusters respectively). Let us consider $M(T_i)$ as the Markovian transition matrix model. In transition from time slot T_i to T_{i+1} , for each time stamp $t_{j'} \in T_{i+1}$, we evaluate corresponding renewed transition probability for changes in

the link structure, but we leave the computation of clusters (except for the starting time stamp of the first slot T_1) and cluster evolution probability till the time-varying graph for last time stamp of T_{i+1} occurs. We consider edge weight, based on the number of interactions (say, number of papers they co-authored, number of times they interacted etc.) over that time slot, to compute the clusters. The transition probabilities are evaluated based on the procedures in the following subsections. The ij^{th} entry of the matrix $M(T_i)$ represent the conditional probability that a node v_i with d_i links will interact with a node v_j with d_j links. We represent model $M(T_i)$ with row and column elements as sub-matrices of clusters, where each sub-matrix has the associated set of vertices in that respective cluster representing row and column indices.

In Figure 1, $C_i^1, C_i^2, \dots, C_i^k$ are the row and column sub-matrices, representing the corresponding clusters in the underlying time-varying network graph. Each sub-matrix has number of rows and columns equal to the number of vertices in the corresponding cluster. Elements in the main diagonal are the prediction probability for future interactions between pair of nodes within any cluster, whereas probabilities in the upper and lower-diagonal matrix elements correspond to pair of nodes in distinct clusters, and are called intra- and inter-cluster link prediction probabilities respectively.

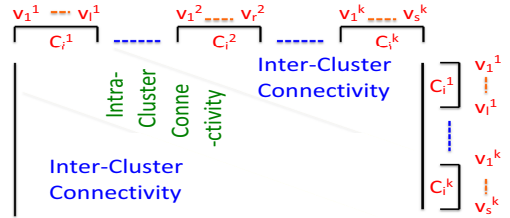


Fig. 1: Markovian Prediction Matrix Model

If a node pair has degree d_i and d_j in cluster C_i^j and C_i^k respectively, then we represent the link probability between them to be $p_{d_i d_j}^{C_i^j C_i^k}$.

Besides the proposed model framework, the mathematical formulae underlying the model, the approach for model parameter evaluation and interaction probability computation, is inspired by intrinsic operations over real world networks. For example, over academic collaboration network, a CS researcher can collaborate with Physicists to have a one time paper in Quantum Computing area, or have a set of papers over time slot(s). Here, the individual-one time collaboration corresponds to links over different time stamps, whereas the later one reflects changed community structure over time slot. Thus, comes the evolution of links over time stamps and evolution of clusters over time slots, as described in Subsection III-B. Further, it is evident from online social networks (e.g. Facebook, Twitter etc.) that in any interaction, nodes that reply, re-tweet etc. are considered initiator of the action. In response, receiving nodes may reply, re-tweet (to others), like a post etc. This type of cause-effect-based interactions or occurrence of pairs of interactions, gives rise to the concept of temporally correlated evolution over both time scales, as is used in Subsections

III-B1, III-B2 and subsequently in III-C. More so, a person with more collaborators or friends gets opportunity for more interactions; similarly persons' active duration attracts others to interact with. In addition, receiver communication with others depends on its set of associates (degree), its duration and interaction frequency. In fact these basic attributes are used in the detailed formulation of Subsections in III-B, and III-C.

2) *Effect of Node Degree in Real World Networks:* Here we give a simple degree based prediction probability computation. In real world application networks, nodes of high degree prefer to interact with other high degree nodes. Though, there are also high degree nodes interacting with other low degree nodes, justifying the right-skewed degree distribution [19] in them. So, to some extent interaction between pair of nodes is dependent on both source and destination node degree or how much they communicate. In fact, the Pearson correlation coefficient of the degrees of the end points of an edge is given by [20]. In order to reflect the explicit correlations we need the distribution of interaction $p(d_i)$ and the conditional probability that a node of degree d_i interacts with a node of degree d_j , i. e., $p(d_j|d_i)$, $\sum_{d_i} p(d_i) = 1$ and $\sum_{d_j} p(d_j|d_i) = 1$. In the simplest Markovian model, each of the matrix elements can be represented as $p(d_i) \times p(d_j|d_i)$.

3) *Initializing Markovian Matrix Elements:* Over time stamp $t_{i'}$ the changes in the elements of M_{T_i} (also in $M(t_{i'})$) are effect of local evolution, where as over time slot T_i it is the effect of global evolution. If we start with the network without any interactions (say at $t = 0$), then interactions between pairs of nodes occur uniformly at random and can be selected likewise. We now discuss the initialization of matrix elements in presence of starting interaction set.

Initializing $M(t_{1'})$ in Presence of Interactions: At time stamp $t_{1'}$ corresponding to T_1 , we have the first time-varying graph and is the start of our observation. At this point we do not have any past interaction information to derive edge weights. So, we use node attributes (say, authors in same research field, same University, and people in the same geographic location, affiliated to same interest/club) to form the initial tentative clusters, and the resulting network graph has now both intra- and inter-cluster interactions. We now compute the probabilities within cluster $p_{d_i d_j}^{C_1^j C_1^j}(t_{1'})$ and across cluster $p_{d_i d_j}^{C_1^j C_1^k}(t_{1'})$. Let v_i, v_j be the vertices with d_i, d_j number of links respectively. When $v_i, v_j \in C_1^j$, let $p(d_i)$ and $p(d_j)$ be their associated degree distributions w.r.t. C_1^j . Thus, we define intra-cluster initial link probabilities as:

$$p_{d_i d_j}^{C_1^j C_1^j} = p(d_i)p(d_j) \frac{d_{in}(v_i)}{\sum_{v \in C_1^j} d_{in}(v)} \times \frac{d(v_j)}{\sum_{v \in C_1^j} d(v)}. \quad (1)$$

In a similar manner, we initialize the initial inter-cluster link probabilities given by the following Equation:

$$p_{d_i d_j}^{C_1^j C_1^k} = p(d_i)p(d_j) \frac{d_{in}(v_i)}{\sum_{v \in C_1^j} d_{in}(v)} \times \frac{d(v_j)}{\sum_{v \in C_1^k} d(v)}, \quad (2)$$

where nodes $v_i \in C_1^j$, $v_j \in C_1^k$; corresponding degree distribution $p(d_i)$ and $p(d_j)$ over modules C_1^j and C_1^k respectively.

The weighted in degree reflects a node's position as a start node, whereas weighted total degree on end node reflects the possibility of a leaf node being connected with zero out degree. In case the underlying clusters are ignored at the start of observation, the node degrees are weighted relative to the whole network instead of their respective cluster.

B. Markov model: Evolution metric Computation

In this Subsection we discuss the procedures for evaluating parameters that are in turn used in computing link prediction probabilities in Subsection III-C.

1) *Correlated evolution of links:* We consider the set of time-varying graphs over time stamps $t_{i'} \in T_j, \forall T_j \in [0, T]$. We compute the correlated evolution of links w.r.t. the set of time-varying graphs over each time stamp. Over subsequent time stamp pairs $t_{i'}, t_{j'} \in T_j$, we have $L(t_{i'})$ and $L(t_{j'})$ as the corresponding link sets over time-varying graphs $G_{t_{i'}}$, $G_{t_{j'}}$ respectively. We determine correlated link evolution over successive time stamps as follows:

$$Corr_{Link}(t_{j'}, t_{i'}) = \frac{|L(t_{i'}) \cap L(t_{j'})|}{|L(t_{i'}) \cup L(t_{j'})|}. \quad (3)$$

Similarly, over pairs of successive time stamps in each time slot we consider average correlated evolution of the corresponding pairs of time-varying graphs. We evaluate average correlated evolution of links over each slot T_j as given by the following Equation.

$$\zeta_{link} = \frac{\sum_{t=t_{1'}}^{t_{k'}} Corr(t, t + \delta t)}{t_{k'} - t_{1'}}, \quad (4)$$

where $\delta t = t_{j'} - t_{i'}$, with $t_{i'}$ and $t_{j'}$ being the two successive time stamps over any time slot T_i . $1 - \zeta_{link}$ is the average number of links changed over pairs of successive temporal graphs in any time slot T_i .

As link evolution can be visualized as a cascaded process that takes place in number of successive steps, we consider pairwise link occurrence rate. Likewise, for each link in a time-varying graph we determine the number of times a link l_j follows a link l_i in $t_{i'}$, i.e. $F_{(l_i, l_j)}(t_{i'})$. For example, let A, B, C be three network nodes. Let, l_i is the contact between A and B; l_j is the contact between B and C. (Say, over Twitter application network, let A tweets a message to B and B tweets to C.) Thus, $F_{(l_i, l_j)}(t_{i'})$ is the number of times B communicates with C, following As' communication with B in $t_{i'}$ (say, a day). The weighted measure of the pairwise link occurrence over a single time-varying graph $G_{t_{i'}}$ is given by:

$$F_{(l_i, l_j)}^{weighted}(t_{i'}) = \frac{F_{(l_i, l_j)}(t_{i'})}{\sum_{\forall l_{i'} \neq l_i} F_{(l_i, l_{i'})}(t_{i'})}, \quad (5)$$

Relative to successive time stamps, the correlated pairwise link occurrence rate over successive time-varying graphs $G_{t_{i'}}$, $G_{t_{j'}}$ is denoted as $F_{Corr}(t_{i'}, t_{j'})$. We define $F_{Corr}(t_{i'}, t_{j'})$ as

$$F_{Corr}(t_{i'}, t_{j'}) = MIN\{F_{(l_i, l_j)}^{weighted}(t_{i'}), F_{(l_i, l_j)}^{weighted}(t_{j'})\}, \quad (6)$$

Similarly, with $\delta t = t_{j'} - t_{i'}$, average correlated pairwise link occurrence is defined as:

$$F_{Corr}^{Avg} = \frac{\sum_{t=t_{1'}}^{t_{k'}} F_{Corr}(t, t + \delta t)}{t_{k'} - t_{1'}}. \quad (7)$$

2) *Correlated evolution over clusters*: We determine the clusters corresponding to each $G_{T_i}, \forall T_i \in \{T_1, \dots, T_k\}$, based on the edge weight over T_i . The 1st step is to create the clusters over respective time slot. As mentioned earlier, after creating the initial cluster at time stamp $t_{i'}$ in time slot $[0, T_1]$, the subsequent clusters are created over T_i instead of $t_{i'}$ s. We use incremental variant of the clustering algorithm in [21]. Over the initial clustering from $t_{i'}$ of T_1 , we use the community formation algorithm of [21] at time T_1 . The resulting clusters refers to the underlying graph structure in G_{T_1} . We form subsequent clusters incrementally over rest of the T_i s. In the incremental version, we compute the relative measure of the interaction weight of the node with other nodes within its cluster, to nodes outside its cluster. To be more precise, the interaction weight of nodes with outside cluster, is considered w.r.t. every other cluster, created from the last time slot. In other words, the relative weight measure is computed with the nodes interaction weight within its cluster to interaction weight with every other cluster, taken the other clusters one at a time of which the interacting nodes are part of. If the interaction weight for both the compared clusters is same, then the node belongs to the cluster with higher number of nodes that it interacted with; else, it joins the cluster with higher interaction weight with it in the current time slot. Incase both weight and number of nodes that it interacted with are same over the different clusters, it remains part of the current cluster. This way the clusters in the successive time slots are formed, but what is important for our work here is, how the clusters are correlated over successive time slots.

We compute correlation coefficient for both within and across cluster. For a cluster C_i^k in time T_i , we evaluate its correlation with two successive clusters C_i^k and C_{i+1}^k in respective time slots T_i and T_{i+1} as, $\frac{|C_{T_i}^k \cap C_{T_{i+1}}^k|}{|C_{T_i}^k \cup C_{T_{i+1}}^k|}$. If more than one cluster merges into a single cluster over successive intervals, or a cluster divides into two or more clusters over successive intervals we compute their joint correlation effect as $\prod_{\forall j} \frac{|C_{T_i}^k \cap C_{T_{i+1}}^j|}{|C_{T_i}^k \cup C_{T_{i+1}}^j|}$, where cluster C_i^k evolves into clusters C_{i+1}^j s or clusters C_{i+1}^j s merge into cluster C_i^k over successive time interval. For any pair of clusters that do not either evolve from or to each other over successive intervals it is simply computed as, $\frac{|C_{T_i}^k \cap C_{T_{i+1}}^j|}{|C_{T_i}^k \cup C_{T_{i+1}}^j|}$. Thus, the total correlated evolution among all pairs of successive clusters is given by:

$$Corr_{Cluster}(T_i, T_{i+1}) = \sum_{k,j} \left[\frac{|C_{T_i}^k \cap C_{T_{i+1}}^j|}{|C_{T_i}^k \cup C_{T_{i+1}}^j|} + \prod_{\forall j} \frac{|C_{T_i}^k \cap C_{T_{i+1}}^j|}{|C_{T_i}^k \cup C_{T_{i+1}}^j|} \right] \quad (8)$$

We evaluate average correlated evolution over successive time slots is given by:

$$\zeta_{cluster} = \frac{\sum_{T_i=T_1}^{T_k} Corr_{Cluster}(T_i, T_{i+1})}{T_k - T_1} \quad (9)$$

In the above, $Corr_{Cluster}(T_i, T_{i+1})$ is evaluated over succes-

sive time slots.

Before we discuss the computation of interaction probabilities, we give a schematic presentation (Figure 2) of the steps involved in the prediction process and their direct (the solid arrows) and indirect (the dotted arrows) inter-relationship.

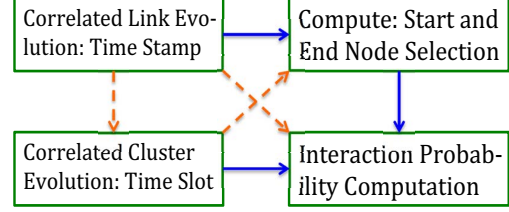


Fig. 2: Schematic Presentation: Inter-relationship Among Prediction Steps

C. Markov model: Computation of Interactions

Here, we describe the steps involved in computing interactions, i.e. procedure for selecting start and end nodes, and computing the interaction probability using the parameters evaluated in the previous Subsection.

1) *Selection of Start and End Nodes*: We give the selection procedure for start and end nodes as follows:

Selection of Start Node (Initiator):

The probability that node v becomes initiator at time $t_{i'+1}$ be represented as $q_{start}^v(t_{i'+1})$. For any node $v \in C_i^j$, v to be an initiator at time $t_{i'+1}$ is dependent on its influence and fraction of its neighbors that are initiators at time $t_{i'}$, the correlated evolution of its outgoing links, and average rate of pairwise link creation.

If $\rho_v(t_{i'})$ is the fraction of start nodes (initiators) incident on node v at time $t_{i'}$. Then $\rho_v(t_{i'}) = \frac{d_{in}(v)}{\sum_{v_i \in C} d_{out}(v_i)}$, evaluated w.r.t. $t_{i'}$, v_i is neighbor of v . Further, for any node v , its influence depends on the relative weight of its degree and distribution of its node degree. Relative weight of node v 's degree at time $t_{i'}$ is $\frac{d(v)}{\sum_{v_i \in C} d(v_i)}$ and its degree distribution is $p(d(v))$. Thus, start node selection probability is computed as follows:

$$q_{start}^v(t_{i'+1}) = \rho_v(t_{i'}) \times \frac{d(v)}{\sum_{v_i \in C} d(v_i)} \times p(d(v)) \times \zeta_{link} \times d_{out}(v) \times F_{Corr}^{Avg} \times |v \in l_i \cap l_j|, \quad (10)$$

where $l_i \cap l_j$ finds the common node between the two links at time $t_{i'}$, the correlated evolution of outgoing links, and average rate of pairwise link creation for node v are $\zeta_{link} \times d_{out}(v)$ and $F_{Corr}^{Avg} \times |v \in l_i \cap l_j|$ respectively. When $d_{out}(v)$ at time $t_{i'}$ is 0 or $|v \in l_i \cap l_j| = \phi$ at time $t_{i'}$, the partial multiplicative terms in the above are substituted with $1 - \zeta_{link}$ and $1 - F_{Corr}^{Avg}$ respectively.

Selection of End Node (Receiver):

Probability of node v to be selected as end node, denoted as $q_{end}^v(t_{i'+1})$, depends on its affinity to start node, the correlated evolution of its incoming links, and average rate of pairwise link creation.

Let $a_{uv}(t_{i'})$ be the affinity between node u and v at time $t_{i'}$, where affinity is inverse of the number of hops between u and v . Thus, the probability of a node being selected as end node given by:

$$q_{end}^v(t_{i'} + 1) = a_{uv}(t_{i'}) \times \zeta_{link} \times d_{in}(v) \times F_{Corr}^{Avg} \times (F(l_i, (u, v)) + F((u, v), l_k)), \quad (11)$$

where (u, v) denotes the respective link and $(l_i, (u, v))$ represent (u, v) follows link l_i . When $d_{in}(v) = 0$, and $(F(l_i, (u, v)) + F((u, v), l_k)) = 0$ at time $t_{i'}$, the partial multiplicative terms in the above are substituted with $1 - \zeta_{link}$ and $1 - F_{Corr}^{Avg}$ respectively. Number of hops is sufficient to limit to five but due to small world concept, during experimental evaluation we find a limit of three hops works fine.

2) *Effect of Node Age and Node Degree*: The interaction of a node for next time interval (say, $t_{i'+1}$) is dependent not only on its duration of presence in the network, but also on its influence/degree in current time. We evaluate a node's duration of presence in the network as $t_p = (t_{i'+1}) - t_a$, where t_a is the time node v start to be part of the network. As $t_{1'}$ is the start of the observation time, the observation duration of the node is $t_o = (t_{i'+1}) - t_{1'}$; thus, the weighted age of the node become $t^{weighted}(v) = \frac{t_p}{t_o}$.

Now, at time $t_{i'}$, the weighted node degree is, $d^{weighted}(v) = \frac{d(v)}{2E_{t_{i'}}}$, where $d(v)$ is node degree and $E_{t_{i'}}$ is total number of edges in the network. To incorporate their mutual effect ψ^{joint} , we consider their joint weighted parameter to predict the occurrence of interaction for node v , thus:

$$\psi^{joint}(v) = t^{weighted}(v) \times d^{weighted}(v). \quad (12)$$

The weighted degree of a node is computed relative to the whole network, instead of its respective cluster.

3) *Global effect in link prediction*: In computing global effect, we consider time slot (i.e. T_i s) as the concerned time interval of our time-varying graph, instead of time stamps. First, we consider global structural parameter like modularity index in link formation. At any time T_i , let the number of edges in a cluster C_i^j be E_i^j , the total number of edges in the network be E_{T_i} , and the number of edges that a cluster C_i^j have with other clusters (C_i^k) is $\sum_{C_i^k} E_i^{jk}, \forall k \neq j$. We compute modularity index [22] (Q_i^j) as, a relative measure of the number of edges within C_i^j and across C_i^j . Thus, $Q_i^j = \frac{E_i^j}{E_{T_i}} - \left(\frac{E_i^{jk}}{E_{T_i}}\right)^2$. We formulate the probability of an edge over nodes $u, v \in C_i^j$ (i.e. $\phi^{C_i^j C_i^j}(uv)$) to be dependent on both modularity index, correlated evolution over clusters and influence of the respective nodes. Thus we have:

$$\phi^{C_i^j C_i^j}(uv) = Q_i^j \times \zeta_{cluster} \times \frac{d(u)}{2E^j} \times \frac{d(v)}{2E^j}. \quad (13)$$

Further, for a pair of nodes across distinct clusters, we

evaluate the probability of a linkage as:

$$\phi^{C_i^j C_i^k}(uv) = (1 - \zeta_{cluster}) \times \frac{E_i^{jk}}{\sum_{\substack{l \neq i \text{ and } m \neq j \\ l \neq j \text{ and } m \neq i}} E_i^{lm}} \times \frac{E_i^{jk}(u, v)}{\sum_{\forall u' \neq u \text{ and } v' \neq v} E_i^{jk}(u', v')}, \quad (14)$$

here on the right hand side, the second term shows the connection across these two clusters relative to other cluster pairs, and the last term represents the number of links between (u, v) relative to other links across the cluster pair.

Concisely, our approach in Section III performs the following steps to compute prediction probability and select suitable links for subsequent state. It initializes the Markovian transition matrix using the approach in Subsubsection III-A3, at the start of the observation ($t_{1'}$), over $G_{t_{1'}} \in G_{T_1}$. For predicting links in the subsequent interval, it selects the start node using the procedure in Subsubsection III-C1 and Subsubsection III-C2; being two independent parameters, multiplying both gives their joint effect in selecting start node. Once we have the start nodes, the possible destination (end) node selection probabilities are calculated as in Subsubsection III-C1. As we have the source and destination node pairs, their interaction probabilities are calculated considering whether they are in the same or different cluster as in Subsubsection III-C3. The elements of the Markovian matrix in Subsubsection III-A3 are then updated with this probability. The interaction takes place across the selected start and destination nodes with interaction probability above certain threshold limit (say, top 30%). The above procedure is iterated for each successive interval till the end of the training period, evaluated links over each interval being considered for training data for subsequent intervals and then for predicting the link occurrence over time slots $[T, T+1]$ of the test set, and also over time stamps in time slot $T+1$. We use top 5%, 10%, 20% of the predicted link probabilities and compared it with actual occurrence of links to evaluate our prediction model.

Every created edge contributes to the degree of respective start and end nodes. Thus, the degree distribution of the application network depends on these created edges. The degree distribution of the real world large scale networks are known to exhibit power law. So, we need to investigate whether the correctly predicted edges by our prediction model, results in power law for degree distribution over the respective application network.

IV. POWER LAW DEGREE DISTRIBUTION OF PREDICTED EDGES BY PREDICTION MODEL

In this section, we show that our proposed approach (model) for node and edge selection, results in power law distribution of the resulting node degree over the network, as is observed in real world networks. In Section III, we integrated temporal dimension in to our parameters. This helps in synchronization of the prediction model with the underlying network evolution. Moreover, in Subsection III-C2, we integrate age of node along

with its degree, in its selection. Now, let us assume that the time interval between successive edge creation of the node, say v , be $\tau_d(v) = t_{d+1}(v) - t_d(v)$, where $(d+1)$ and d are its node degree at time $t+1$ and t . The closest replication model (distribution) of this time interval is given by the power law with exponential cut off [20], $P_l(\tau_d; \rho, \sigma) \propto \tau_d^{-\rho} \exp(-\sigma \tau_d)$, where d is current degree, ρ, σ are parameters of distribution of τ_d as a function of current node degree d . In Subsection III-C2, we also integrate node degree that reflects the preferential attachment property exhibited by networks. Thus, the slope ρ of the power law distribution remains constant, whereas parameter σ representing exponential cutoff part becomes faster. Thus, successive edges are added at a faster rate, i.e., $(d+1)^{st}$ edge of a node added faster than d^{th} edge. So, distribution of τ_d becomes $P_l(\tau|d; \rho, \sigma) \propto \tau^{-\rho} \exp(-\sigma d \tau)$. A degree d node v , samples with intervals τ from $P_l(\tau|d; \rho, \sigma) = \frac{1}{Z} \tau^{-\rho} \exp(-\sigma d \tau)$ and remains inactive for a period τ . Additionally, the experimental analysis of nodes life time in Section V shows that they are exponentially distributed with parameter say, λ .

Theorem IV.1. The resulting node degree distribution exhibits power law.

Proof. We evaluate the normalization constant Z from the distribution for time interval $P_l(\tau|d; \rho, \sigma) = \frac{1}{Z} \tau^{-\rho} \exp(-\sigma d \tau)$. We have, $Z = \int_0^\infty \tau^\rho e^{-\sigma d \tau} d\tau = \frac{\Gamma(1-\rho)}{(\sigma d)^{1-\rho}}$. Let T be the life time of a node. Between two successive samples of a node from the distribution $P_l(\cdot)$, it (node) remains inactive for $\tau(k)$ time period. Each successive sampling increases node degree by 1. Let D be the final degree a node attained in its life time T . In attaining D , the node remains inactive in total for:

$$\sum_{k=1}^D \tau(k) \leq T. \quad (15)$$

On a similar note, the expected time interval $E(\tau|d; \rho, \sigma)$ for a node of degree d is:

$$E(\tau|d; \rho, \sigma) = \frac{\Gamma(2-\rho)}{\Gamma(1-\rho)} (\sigma d)^{-1}. \quad (16)$$

From Equations (15) and (16), we relate lifetime T and expected final degree D of the node to have inequality:

$$\sum_{d=1}^D \frac{\Gamma(2-\rho)}{\Gamma(1-\rho)} (\sigma d)^{-1} = \frac{\Gamma(2-\rho)}{\Gamma(1-\rho)} (\sigma)^{-1} \sum_{d=1}^D d^{-1} \leq T. \quad (17)$$

We know that $\sum_{d=1}^D d^{-1} = \Theta(\ln D)$. From Equation (17), the final degree D of a node with lifetime T is:

$$D \approx \exp\left(\frac{\Gamma(1-\rho)}{\Gamma(2-\rho)} \sigma T\right) \quad (18)$$

Thus, D is exponential function of the lifetime (T) of a node. We can write it as, $D = \exp(\varphi T)$, where $\varphi = \frac{\Gamma(1-\rho)}{\Gamma(2-\rho)} \sigma$.

Further, the node life time is exponentially distributed; thus, the distribution of node degree (D) using λ and φ will be:

$$D \sim \frac{\lambda}{\varphi D} e^{-\frac{\lambda}{\varphi} \log D} = \frac{\lambda D^{-(1+\frac{\lambda}{\varphi})}}{\varphi}.$$

With exponent $(1 + \frac{\lambda}{\varphi})$, the node degree is power law

distributed. Thus, predicted edges in our model results in power law degree distribution, as in case of real world networks. $\square$

V. EXPERIMENTAL RESULTS

We evaluate our approach using real world data sets and comparing the prediction accuracy over the standard receiver operating characteristic (ROC) curve with false positives in X -axis and true positives in Y -axis to represent the outcomes in successive time intervals. The area under the ROC curve (AUC measure) represents the achievable true positive rate (TPR) relative to false positive rate (FPR). The ROC curve helps us evaluate the effectiveness of prediction model from tradeoff between TPR and FPR. The higher the TPR than FPR, the better the model; though, in general $TPR > FPR$ is considered a good model.

We evaluate and compare the accuracy of our Markov model based approach (Das) relative to the state-of-the-art approaches such as: Adamic-Adar score (AA), Katz-measure, Common Neighbors (CN) mentioned in [12] and the recent dynamic approaches such as Sarkar et al. [16] (Sarkar) and Tyenda et al. [15] (Tyenda). The AA, CN, Katz are static approaches based on the aggregated graph over $[0, T]$. TPR and FPR are computed from fraction of correct predictions and fraction of wrong predictions. The measure is equally applicable to evaluate prediction accuracy of selected start nodes. Further, the AUC values resemble ROC curve, in reflecting the performance/accuracy of the approach.

We use Twitter, Enron and DBLP dataset. Use of Enron and DBLP data set help effective comparison with the work of Sarkar et al. [16] and Tyenda et al. [15], as they also use the same data sets. Further, Twitter network data set helps us evaluate the effectiveness of all the dynamic approaches. The DBLP dataset is from [23], to which the year of collaboration timestamps are attached and 3000 authors with at least 5 papers are considered. With Twitter API, over 88,930 time stamped interactions from reply, retweet, favorite message during a period of 32 weeks are collected from Twitter network (July, 2015-to-January, 2016). In the Enron data set [24], the email communication in the range of July 2000 to March 2002 is considered for our experiments.

Over Enron and twitter, the data is time stamped in hour:min format. A day is taken as the smallest time unit i.e. time stamp over these networks, whereas a month and a week taken as time slots over them, respectively. For DBLP network, we consider a year as the time slot and time stamps are half yearly. The frequency of interaction between node pairs is incorporated over the respective time stamps.

We consider all time stamps in one slot to predict over next time stamp and all time slots to predict over next slot; thus, the number of history states s equals the number of time stamps or total number of time slots in respective time domain. For example, in twitter application network, we consider time stamps over three slots. Thus, $s = 21$ for time stamps, and $s = 31$ for time slots, used for prediction over next time stamp or time slot. Further, over Enron email application network

we have $s = 20$ w.r.t. time slots, whereas, $s = 30$ over time stamps.

In Figure 3, the age of nodes over three networks follow exponential distribution, as the elongated shape along X and Y-axis for each is observed. This justifies our consideration of exponential parameter λ in Section IV. Further, the inter-arrival time between interaction is power law distributed (exponent between 2-to-3) with exponential cut off over all the three networks, as is evident from the Figure 4. Moreover, the parameters ρ associated with their distribution remains almost constant (Y-value almost same throughout per curve) with degree d , where as the other associated parameter σ grows linearly with time, as is reflected in Figure 5. We used these facts in Section IV. In Figure 3 and Figure 4, we presented separate graph for each network owing to distinct ranges.

First, we show the accuracy of our start node selection with top 30% of the computed probabilities over DBLP, Twitter and Enron datasets in Figure 6. To clarify further, when we use top 30% of the link prediction probabilities, we check the TPR and FPR values within this set of predicted links. Owing to the relatively lesser dynamic (time-varying) nature of the DBLP dataset, it has better accuracy than other two datasets.

Now, we evaluate the prediction accuracy over the next time slot, i.e. $[T, T + 1]$. The ROC curves in Figure 7 and the 2nd graph of Figure 6 presents the prediction accuracy of our approach (Das) relative to the other approaches over three datasets. The three dynamic approaches (Tylanda, Sarkar and Das) outperform the remaining static approaches, as is evident from the ROC curves (and AUC values Table II). Further more, the probabilistic Markov model in (Das) has higher prediction accuracy than the integration of local network structure and time series based approaches (Tylanda and Sarkar). The better prediction accuracy of our model is attributed to the two way temporal analysis, that is consideration of two time scales and the corresponding local and global structural evolution. Here, we use top 20% of the predicted links in all cases, for evaluating accuracy of the proposed model. Each of the ROC curves shows the variation of rate of correct predictions to rate of wrong prediction.

In the 1st graph of Figure 8, we show the link prediction accuracy while selecting top 5%, 10%, 20% of the prediction probabilities for links over Twitter network. In the 2nd graph we show the prediction accuracy (with top 20% and 30% link selection) over time stamps of one slot ($[T, T + 1]$). Here, we have included the effect of cluster evolution and rate of evolution computed over time slots, to predict for the next time stamps (days in Twitter network). Thus, prediction accuracy (with 20%) for time stamps and slots are almost similar over Twitter network.

To avoid possible over fitting in the above predictions, instead of taking only $[0, T]$ as training set and $[T, T + 1]$ as test set, we rather consider k -fold cross validation. To compute the prediction accuracy of model, we iterate it k times, over our observation interval and take average of the prediction outcomes over all test intervals. For example: in

Twitter application network, of the 32 week (slots) observation period, we take sets of 31 weeks as training set in each iteration and predict over the remaining slot in that iteration. Over 32 test sets, we take average of the prediction outcome to determine the final prediction accuracy of the model. Table III shows the resulting prediction accuracy of three dynamic models, over three distinct datasets, with higher accuracy of our proposed model (Das). In Figure 9 we show the corresponding ROC curves for Enron and Twitter networks. Here, we show only the dynamic models, as static models have further lower accuracy.

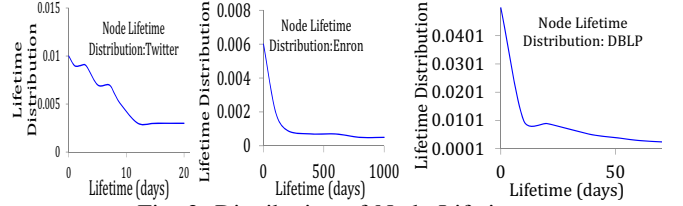


Fig. 3: Distribution of Node Lifetime

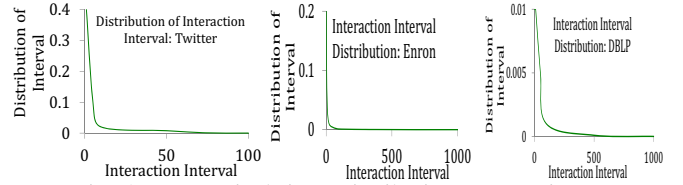


Fig. 4: Inter-arrival time Distribution: Interactions

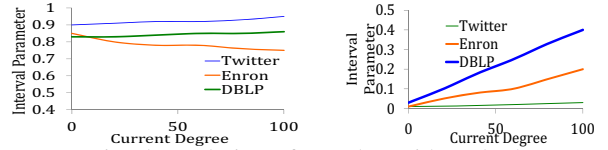


Fig. 5: Evolution of ρ and σ with node degree

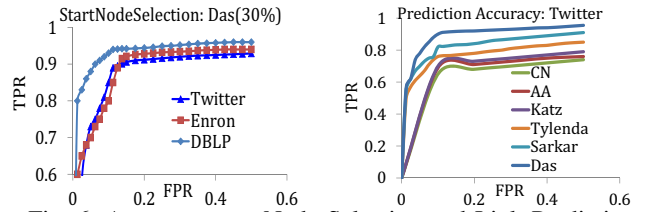


Fig. 6: Accuracy over Node Selection and Link Prediction

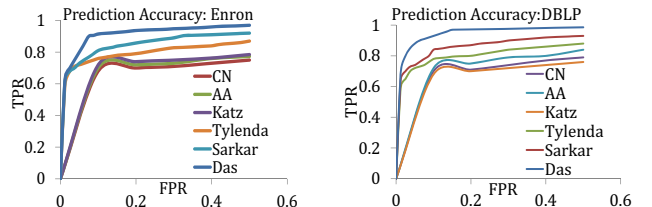


Fig. 7: Accuracy over Link Prediction

It is interesting to note that the ROC curves for Figure 7 and 2nd graph of Figure 6, seems to have a lesser TPR around FPR value of 0.1 (we do not give any smoothed version of ROC

curves, so at some points we may see this), for all baseline approaches (CN, AA, and Katz); but, the curves have an overall upward trajectory. Further, the ROC curves in Figure 9 have relatively lesser prediction accuracy than the earlier curves, without over fitting being taken care of.

TABLE II: Average AUC value with Our model

AUC results over Twitter, Enron, DBLP datasets						
Dataset	CN	AA	Katz	Tylenda	Sarkar	Das
Twitter	0.6467	0.6816	0.6968	0.7643	0.8293	0.9183
Enron	0.6615	0.6753	0.7389	0.7835	0.9031	0.9328
DBLP	0.7523	0.7721	0.7338	0.8230	0.9143	0.9613

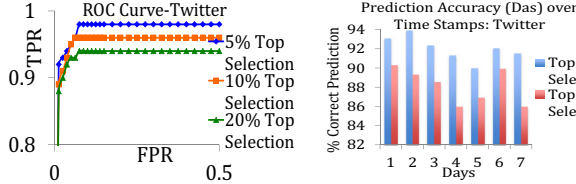


Fig. 8: Link Prediction Accuracy: TimeSlot and TimeStamp

TABLE III: Average AUC value with Our model

AUC results over Twitter, Enron, DBLP datasets			
Dataset	Tylenda	Sarkar	Das
Twitter	0.7039	0.7935	0.8893
Enron	0.7313	0.8189	0.9103
DBLP	0.7953	0.8456	0.9319

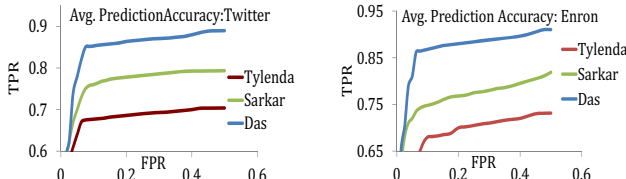


Fig. 9: Average Link Prediction Accuracy Over All Slots

VI. CONCLUSION

Not surprisingly predictions models need to consider history to have better prediction accuracy and link prediction models are no exception. The most significant challenge is to integrate the time-varying networks and underlying network structural dynamics into the prediction model. The key to mitigate this challenge and achieve best possible prediction accuracy is to consider the time-varying nature of the network over microscopic and macroscopic time scales, and integrate the desired structural dynamics over respective time domains. Thus, justifying the superiority of our proposed model over existing models.

To keep our focus on prediction accuracy of the proposed model, we exclude the discussion on computational cost and scalability of the approach. In fact both are inter-related and the cost analysis is dependent on the upper bound on the order of Markov model. The order of the Markov model is the number of history states to be kept track of over respective time scales. We propose to give a theoretical upper bound on the number of history states using compression theory and use the computation cost per state, to derive the computation cost

of the proposed model. We will also support it with extensive experimental results.

ACKNOWLEDGMENT

This work is partially supported by NSF grants under award numbers CCF-1533918 and CNS-1355505.

REFERENCES

- [1] D. Kempe, J. Kleinberg, and A. Kumar, "Connectivity and inference problems for temporal networks," in *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, 2000, pp. 504–513.
- [2] V. Nicosia, J. Tang, C. Mascolo, M. Musolesi, G. Russo, and V. Latora, "Graph metrics for temporal networks," in *Temporal Networks*. Springer, 2013, pp. 15–40.
- [3] P. Holme and J. Saramäki, "Temporal networks," *Physics reports*, vol. 519, no. 3, pp. 97–125, 2012.
- [4] S. F. Adafre and M. de Rijke, "Discovering missing links in wikipedia," in *Proceedings of the 3rd ACM International Workshop on Link Discovery*, 2005, pp. 90–97.
- [5] B. Taskar, M.-F. Wong, P. Abbeel, and D. Koller, "Link prediction in relational data," in *Advances in Neural Information Processing Systems*, 2003.
- [6] C. A. Bliss, M. R. Frank, C. M. Danforth, and P. S. Dodds, "An evolutionary algorithm approach to link prediction in dynamic social networks," *Journal of Computational Science*, vol. 5, no. 5, pp. 750–764, 2014.
- [7] C. C. Aggarwal, Y. Xie, and P. S. Yu, "A framework for dynamic link prediction in heterogeneous networks," *Statistical Analysis and Data Mining: The ASA Data Science Journal*, vol. 7, no. 1, pp. 14–33, 2014.
- [8] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on opportunistic forwarding algorithms," *IEEE Transactions on Mobile Computing*, vol. 6, no. 6, pp. 606–620, 2007.
- [9] Y. Cao, X. Chen, T. Jiang, and J. Zhang, "Socast: Social ties based cooperative video multicast," in *Proceedings of The IEEE INFOCOM*, 2014, pp. 415–423.
- [10] E. Richard, S. Gaïffas, and N. Vayatis, "Link prediction in graphs with autoregressive features," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 565–593, 2014.
- [11] M. Al Hasan and M. J. Zaki, "A survey of link prediction in social networks," in *Social Network Data Analytics*. Springer, 2011, pp. 243–275.
- [12] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 7, pp. 1019–1031, 2007.
- [13] A.-L. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, pp. 509–512, 1999.
- [14] R. Sarukkai, "Link prediction and path analysis using markov chains," *Computer Networks*, vol. 33, no. 1, pp. 377–386, 2000.
- [15] T. Tylenda, R. Angelova, and S. Bedathur, "Towards time-aware link prediction in evolving social networks," in *Proceedings of the 3rd Workshop on Social Network Mining and Analysis*. ACM, 2009, p. 9.
- [16] P. Sarkar, D. Chakrabarti, and M. Jordan, "Nonparametric link prediction in dynamic networks," *arXiv preprint arXiv:1206.6394*, 2012.
- [17] S. Hanneke, W. Fu, E. P. Xing *et al.*, "Discrete temporal models of social networks," *Electronic Journal of Statistics*, vol. 4, pp. 585–605, 2010.
- [18] M. Kolar, L. Song, A. Ahmed, and E. P. Xing, "Estimating time-varying networks," *The Annals of Applied Statistics*, pp. 94–123, 2010.
- [19] A. Clauset, C. R. Shalizi, and M. E. Newman, "Power-law distributions in empirical data," *SIAM review*, vol. 51, no. 4, pp. 661–703, 2009.
- [20] M. E. J. Newman, *Networks: An Introduction*. New York, NY, USA: Oxford University Press, Inc., 2010.
- [21] S. Das and S. K. Das, "Leveraging network structure in centrality evaluation of large scale networks," in *40th IEEE Conference on Local Computer Networks (LCN)*, 2015, pp. 579–586.
- [22] M. E. Newman, "Mixing patterns in networks," *Physical Review E*, vol. 67, no. 2, p. 026126, 2003.
- [23] H. Deng, J. Han, B. Zhao, Y. Yu, and C. X. Lin, "Probabilistic topic models with biased propagation on heterogeneous information networks," in *Proceedings of the 17th ACM SIGKDD*, 2011, pp. 1271–1279.
- [24] "Enron e-mail data set," <https://www.cs.cmu.edu/~Jenron/>, accessed: 2016-01-15.