Graph Embedding Framework for Link Prediction and Vertex Behavior Modeling in Temporal Social Networks

Chunsheng Fang$^{1,3}$ Mojtaba Kohram$^1$ Xiangxiang Meng$^{2,3}$ Anca Ralescu$^1$
1 Dept of Computer Science, University of Cincinnati, Ohio, USA, 45221
2 Dept of Mathematical Sciences, University of Cincinnati, Ohio, USA, 45221
3 Cincinnati Children's Hospital Medical Center, Cincinnati, Ohio, USA, 45229
fangcg@mail.uc.edu kohramma@mail.uc.edu mengxa@mail.uc.edu Anca.ralescu@mail.uc.edu

ABSTRACT
We present a novel framework in which the link prediction problem in temporal social networks is formulated as trajectory prediction in a continuous space. Four major modules constitute this framework: (1) graph embedding: the discrete space of graphs is mapped into a continuous space while preserving distances for a given graph kernel; (2) manifold alignment: graph embeddings corresponding to different time points are aligned to achieve low variance trajectories for a more reliable prediction; (3) trajectory prediction: the temporal graphs form a time series, to which various prediction models (e.g. regression) can be applied; (4) graph reconstruction from the predicted graph embedding which is invariant against scale, translation and rotation. Furthermore, this framework enables an innovative way to analyze temporal graph vertex behaviors and visualization. Extensive preliminary results on real world data sets demonstrate the promises of this proposed approach.

Keywords
Link prediction, social network, algorithmic framework, graph embedding.

1. INTRODUCTION
With the recent advent of large-scale social networks and biological networks etc, link prediction in temporal networks emerges as a more and more important research problem. Several survey papers have summarized some recent progress in link prediction [1] and link mining [2]. A thorough survey paper [1] summarizes various “static” methods that explore different graph distance metrics on one network snapshot and try to predict the next one.

Alternatively, recent research focuses more on formulating the link prediction problem in a “dynamic” way, for example, as a time series regression model to accommodate historical data [2,3]. This is done in a more general framework by modeling the whole graph historical dynamics to provide a more insightful understanding into the link prediction problem.

The first important issue in developing such a framework is to decide exactly what needs to be extracted from network such that (I) it can be effectively tracked, and predicted in time, and (II) to use the predicted structure to (re)construct a network state consistent with the network evolution. For example, various features, such as node degrees of the graph representing the network can be used in conjunction with an iterative regression solver to predict the graph features as the network evolves in time [3]. Alternatively, spectral approaches attempt model graph evolution using polynomial curves [4,5]. These assume eigenvectors as stable during time. Finally, a combined time series ARMA model and low rank approximation approach for estimating the eigenvectors of the Laplacian matrix from each time point has been proposed in [6, 13].

In this paper, we revisited most of the existing link prediction algorithms and the recent attempts in temporal social networks, to answer the following important questions which will lead to our framework: What key components contribute to a successful link prediction model? What are their contributions to a reliable model estimation and prediction?

As summarized in [1,9], graph distance metric is essential in encoding the local and global consistency, but simply using it is not sufficient to capture evolutions along time. In order to utilize historical network snapshots under a general framework, a common approach is to convert the discrete graphs into the continuous space while preserving distance constraints among vertices, which can be achieved by graph embedding. Most of the existing continuous link prediction algorithms are actually different derivations of graph embedding. MDS [12] using principal component of the distance matrix; Spectral embedding [10] using eigenvectors of the Laplacian matrix; graph feature tracking [3] using linear embedding defined by a linear mapping.

To ensure a reliable model for estimation and prediction the stability graph-valued time series must be guaranteed. In this paper we submit that the stability aspect did not attract sufficient attention in previous literature. By contrast, in the framework proposed here, we point out that reliability of prediction is achieved only when a key graph feature is preserved in the process of graph embedding.
Since the graph embedding results in a nonlinear subspace, we refer to this constraint as manifold alignment. More precisely, each graph embedding can be viewed as a smooth manifold in the higher dimensional space.

However, the individual graph geometry such as position, orientation, scale, can be arbitrary, making the link prediction problem difficult. The manifold alignment, i.e. alignment of the manifolds corresponding to different time points, will result in smaller variance and more stationary time series. This in turn will contribute to more accurate estimation and prediction.

To address all these above, we propose a general framework as in Figure 1 which formulates this problem as trajectory prediction in a continuous graph manifold space, and decompose it into four essential components: graph embedding, manifold alignment, trajectory prediction, graph reconstruction.

Extensive experiments in real-world social networks evaluate the effectiveness of each component, which demonstrate the promise of our framework for link prediction. Furthermore, several functionalities derived from our framework, such as visualization of temporal social networks as trajectories, vertex behavior modeling, are demonstrated.

2. PROPOSED FRAMEWORK
2.1 Problem Formulation
Following [6], we start by formally defining the problem as follows. A dynamic graph with horizon \( T \), \( \mathcal{G}_T \), is a collection of graphs, \( G_t \), indexed by time, \( t=1,2,\ldots, T \). In this work we consider that the set of nodes remains constant over time. More precisely we define

\[
\mathcal{G}_T = \{ G_t | G_t=(V, E_t), t=1,\ldots,T; E_t \subseteq E_{t+1}, t=1,\ldots,T-1 \}
\]

The link prediction problem is then to predict \( G_{T+1} \), which is actually \( E_{T+1} \), based on the \( \mathcal{G}_T \).

The proposed framework consists of the following:

2.2 Graph Embedding
Graph embedding is the first component of our framework. The goal of the graph embedding component is to embed the graph as a point cloud in some continuous space where its evolution can be tracked using conventional regression techniques. To achieve this goal many of the available methods in Multi-Dimensional Scaling (MDS) can be used.

MDS is generally used to achieve a visualization of a dissimilarity matrix in some smooth space. A good review of the available methods and techniques can be found in [12]. MDS methods can be classified into metric and non-metric methods. In metric MDS the dissimilarity matrix must represent a metric on the embedding space. The MDS algorithm used here will be classical metric MDS where the metric will be the shortest distance between two nodes in the original graph.

After this step, each graph snapshot \( G_t \) is optimally embedded into real numbers \( X_t \) with dimension of \( M \) vertices by \( K \) MDS dimensions, as (1).

\[
X_t \in \mathbb{R}^{M \times K}
\] (1)

2.3 Manifold Alignment
Manifold alignment finds the correspondence between two seemingly disparate datasets [8]. It constitutes a critical component in our proposed framework, due to the variability of manifolds along time points. To see the effect of the alignment, as shown in Figure 2, we consider three alignment procedures: (i) Affine transformation alignment

![Evolution of graph manifolds along time](image)
to the previous graph in the sequence, (ii) Affine transformation alignment to the 1st graph, and (iii) Procrustes alignment the 1st graph.

Procrustes alignment [8] seeks the isotropic dilation and the rigid translation, reflection and rotation needed to correspond two embeddings X and Y, by optimizing the following objective function:

$$Q_{optimal} = \text{argmin}_Q \| X - kYQ \|_F$$  \hspace{1cm} (2)

where \( k = \text{tr}(\Sigma)/\text{tr}(Y^2 \Sigma) \). \( \Sigma \) is the diagonal matrix of SVD of \( Y^2 X \).

With the Procrustes alignment matrix \( Q \), correspondence between two graph manifolds can be performed with respect to different procedures. We denote the aligned manifold in time point \( t \) as:

$$Z_t \in \mathbb{R}^{M \times K}$$  \hspace{1cm} (3)

### 2.4 Trajectory Estimation and Prediction

Two standard regression models, linear and quadric, are employed for both trajectory estimation and prediction of the embedded vertex manifold coordinates. Each vertex's coordinate time series are centered and then fitted together into a nested random effect regression model, as follows:

**Linear:**

$$Z_t(m, k) = \beta_{mk}t + \epsilon_{tmk}$$  \hspace{1cm} (4)

**Quadratic:**

$$Z_t(m, k) = \beta_{mk}t^2 + \gamma_{mk}t + \epsilon_{tmk}$$  \hspace{1cm} (5)

for snapshots \( t = 1, 2, \ldots, T \), authors \( m = 1, 2, \ldots, M \), and dimensions \( k = 1, 2, \ldots, K \). Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC) serve as likelihood criteria for models selection. To compare and select alignment methods, we define the learning errors as the Euclidean distance between the ground truth and the learned positions of a vertex.

$$e_{mt} = \sqrt{\sum_{k=1}^{K} (Z_t(m, k) - Z_0(m, k))^2}$$  \hspace{1cm} (6)

Furthermore, we propose to use a weighted learning error:

$$e_{mt}^w = \frac{1}{\sum_{k=1}^{K} Z_0(m, k)^2} e_{mt}$$  \hspace{1cm} (7)

which is more robust to the random errors on the boundaries of the whole network and put more penalty to the prediction algorithm that have relatively worse performance inside the center of the whole data clusters.

### 2.5 Graph Reconstruction

Prior to this step, an optimal predicted graph embedding \( X_{T+1} \) for \( G_{T+1} \) has been computed. To reconstruct the graph, we need to ensure that the graph kernel from \( X_{T+1} \) is Positive-Semi Definite (PSD) [9]. In our framework, we circumvent this potential issue by building the graph from pair-wise distance in \( X_{T+1} \), which is guaranteed to be PSD.

### 2.6 ALGORITHM

**Input:** A dynamic graph, \( \{G_t, t = 1, 2, \ldots, T\} \).

**Output:** Predicted \( G_{T+1} \).

1. Define a graph distance or kernel; Any PSD graph distance matrix or any Mercer kernel can be adopted. Local and global distance constraints are preserved in this step. Shortest path kernel is adopted in this paper;
2. Graph Embedding. For each graph, apply Multi Dimension Scaling, or other graph embedding algorithms (e.g. using graph spectral) to map the graph into Euclidean space (or other continuous space);
3. Manifold Alignment. Aligned the each \( G_t \) in the embedded space with different choices of alignment algorithms as in Section 2;
4. Trajectory Prediction. For each graph vertex in the embedded space, its trajectory during time can be modeled using any time series regression model, e.g. linear model[7], ARMA, etc. After this step, the graph embedding \( X_{T+1} \) is optimally predicted;
5. Graph Reconstruction. The predicted graph \( G_{T+1} \) can be constructed from the pair-wise distances in the embedded space \( X_{T+1} \).

### 3. EXPERIMENT EVALUATION

The approach described above was implemented on a data set extracted from the DBLP authorship database1. This data set is analyzed from 1995 to 2004. An author is included in this set if he/she contributed in at least eight of the ten years. From this set of authors the CORE set is selected to be the largest connected component of the first snapshot (1995). This results in a CORE set of 2,538 authors at 10 different time points.

### 3.1 Validating Manifold Alignment

Our framework provides an innovative visualization of temporal social networks as Figure 3 illustrates. Each vertex can be viewed as a trajectory along time axis, while each layer represents the graph embedding in each year.

Three important observations can be made from Figure 3:

1. Of all trajectories evolving with time, those without manifold alignment have high temporal variance with large fluctuations;
2. All three alignments have reduced the trajectory variance with different levels;
3. Procrustes alignment performs the best both in magnitude and fluctuation of variance.

---

1 http://dblp.uni-trier.de/
3.2 Gravitational collapse of trajectories
Another interesting phenomenon observed from Figures 3 and 4 (which is not accounted for by previous methods), is that of gravitational collapse of trajectories.

In the current formulation of the link prediction problem, the dynamic graph has a fixed vertex set. The edge set increases in time as edges are never deleted (for the DBLP example, this means that if two authors have been linked, they remain linked). It then follows that the diameters of successive graphs (successive network snapshots) are eventually decreasing as the number of shortest paths increases. The graph embedding and the trajectories of each vertex reflect this property. By analogy to the astronomical gravity effect which attracts the mass and eventually every atom will collapse into a singularity, we call this gravitational collapse of trajectories. In the social networks context this corresponds to the convergence of the graphs representing the network to a complete graph.

3.3 Trajectory Modeling Results
Both linear and quadratic regressions are applied for the estimation and prediction of the author coordinates in year 2004. We fit the model with the first two primary MDS dimensions. For each type of aligned data, the linear fitting has better AIC and BIC scores. Table 1 summarizes the means and the standard deviations of the learning errors for the four types of alignment algorithms. The Procrustes Alignment yields the smallest mean weighed error. It is not surprised that the data without any alignment has errors significantly larger than those from alignment.

Figure 5 presents the scatter plots of ground truth coordinates vs. estimated and predicted coordinates for the network in 2004. Prediction has wider variance than estimation, due to the exclusion of the last year data. Alignment plays a significant role in modeling the dynamic graph. The Procrustes alignment method with linear random effect regression performs well in both estimation and prediction, which strongly supports the claim that under this framework, the nature of dynamic social network is sufficiently captured by a simple model.
Table 1. Estimation and Prediction Errors for the four alignments in linear or quadratic nested random effect models.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Without manifold alignment</td>
<td>Linear</td>
<td>192926.1</td>
<td>192942.2</td>
<td>1.7473(1.0435)</td>
<td>1.2271(0.7722)</td>
<td>2.3604(1.4096)</td>
<td>1.6576(1.0432)</td>
</tr>
<tr>
<td></td>
<td>Quadratic</td>
<td>209312.2</td>
<td>209328.1</td>
<td>1.1329(0.6530)</td>
<td>0.8544(1.9637)</td>
<td>2.5351(1.4613)</td>
<td>1.9120(4.3943)</td>
</tr>
<tr>
<td>Affine transform to previous year</td>
<td>Linear</td>
<td>184649.8</td>
<td>184665.8</td>
<td>0.4283(0.2988)</td>
<td>0.3353(0.5095)</td>
<td>0.5785(0.4036)</td>
<td>0.4529(0.6883)</td>
</tr>
<tr>
<td></td>
<td>Quadratic</td>
<td>203406.2</td>
<td>203422.0</td>
<td>0.6214(0.4009)</td>
<td>0.5524(2.2990)</td>
<td>1.3905(0.8972)</td>
<td>1.2361(5.1446)</td>
</tr>
<tr>
<td>Affine transform to the 1st year</td>
<td>Linear</td>
<td>161244.7</td>
<td>161260.7</td>
<td>0.4043(0.3362)</td>
<td>0.2767(0.6975)</td>
<td>0.5461(0.4541)</td>
<td>0.3738(0.9423)</td>
</tr>
<tr>
<td></td>
<td>Quadratic</td>
<td>183140.3</td>
<td>183156.2</td>
<td>0.4938(0.3750)</td>
<td>0.3702(2.0993)</td>
<td>1.1051(0.8392)</td>
<td>0.8283(4.6977)</td>
</tr>
<tr>
<td>Procrustes alignment</td>
<td>Linear</td>
<td>162879.5</td>
<td>162895.6</td>
<td>0.4071(0.3324)</td>
<td>0.2716(0.6748)</td>
<td>0.5500(0.4491)</td>
<td>0.3669(0.9116)</td>
</tr>
<tr>
<td></td>
<td>Quadratic</td>
<td>184128.5</td>
<td>184144.3</td>
<td>0.5073(0.3750)</td>
<td>0.3677(1.7034)</td>
<td>1.1352(0.8390)</td>
<td>0.8227(3.8117)</td>
</tr>
</tbody>
</table>

Figure 5. The scatter plots of the estimated and predicted coordinates vs. the true coordinates for the DBLP data in 2004.
Top: Estimations of the core author coordinates in 2004 using the whole time series. Bottom: Predictions of the core author coordinates in 2004 using whole time series except 2004. Prediction has wider variance than estimation. Alignment plays a significant role in modeling the dynamic graph, especially Procrustes alignment.
3.4 Vertex Behavior Modeling

Another functionality enable by this framework is vertex behavior modeling, by clustering the trajectories as shown in Figure 6. Using K-means clustering, we compute 10 clusters which grouped authors with similar temporal behaviors together. Similar trajectories indicate the co-evolution patterns of authors.

To get more insights into the clustering, Figure 7 takes a closer look into seven authors’ trajectories with highest vertex degrees. Interestingly, five of them have similar trajectories, and they turn out to be all from Israel, compared to the rest two are from different institutes.

3.5 Reconstructing the Predicted Network

To reconstruct the predicted network we take the following steps:

4. Predict the graph embedding for the last time point;
5. Collect those pairs of vertices which are not connected by an edge in the graph at the preceding time;
6. Sort the vertices collected at the previous step by their distance: pairs of vertices that are closest in Euclidean space could be potential edges;
7. Prioritize links based on the distances computed above.

The reconstruction results are compared against a set of edges created randomly as shown in Table 2. The results of each embedding method with regression methods are reported in terms of how much better they are than a random guess. The percentage of a correct random guess of an edge existence is 0.013%. The consistent superiority of our framework compared random guess suggests that the trend of graph evolution is meaningfully captured. Lastly, the Procrustes alignment method along with a linear regression method proved to be the best performing predictors, which is consistent with Section 3.4.

4. CONCLUSIONS AND FUTURE WORK

We have innovatively formulated the problem of link prediction in a network which evolves over time. We

---

**Table 2. Performance of the algorithm with different alignment and regression methods compared to random**

<table>
<thead>
<tr>
<th></th>
<th>Without manifold alignment</th>
<th>Affine transform to previous year</th>
<th>Affine transform to the 1st year</th>
<th>Procrustes alignment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Linear</td>
<td>Quadratic</td>
<td>Linear</td>
<td>Quadratic</td>
</tr>
<tr>
<td>Factor of improvement over random</td>
<td>4.49</td>
<td>3.91</td>
<td>4.57</td>
<td>4.78</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3.93</td>
<td>3.68</td>
</tr>
</tbody>
</table>
started from the premise that evolution in time of the network requires a dynamic approach, which take into account this evolution (as opposed to approaches based on node similarities in a static, snapshot of the network). This leads us to consider time series approach. However, the challenge for this approach was to extract suitable, useful network characteristics. This was done by embedding the graphs underlying the network (at each time moment) into a continuous space, resulting in a nonlinear subspace or manifold. Essentially for this approach, ensuring reliable prediction and estimation is the step of manifold alignment. Experimental results support this approach both in (i) the need of the alignment, and (ii) estimation and prediction reliability.

This proposed framework identifies key components in constructing a good link prediction model, but has not been thoroughly exploited all its potentials. What combinations of these key components will theoretically guarantee a good link prediction result? Is there a theoretical upper or lower bound for link prediction upon this framework? These question remains as future work.

5. ACKNOWLEDGMENTS
We appreciate the reviewers’ precious comments.

6. REFERENCES