Graph Spectra Regression with Low-Rank Approximation for Dynamic Graph Link Prediction*

Chunsheng Fang, Jason Lu, Anca L. Ralescu,
Department of Computer Science, University of Cincinnati,
Cincinnati, OH, 45221-0030,
fangcg@mail.uc.edu, long.lu@cchmc.org ,anca.ralescu@uc.edu

Abstract

We propose a temporal regression model for dynamic graph link prediction problem, under spectral graph theory and low rank approximation for the graph Laplacian matrix. Link prediction is important in large-scale graphs including social networks, biological networks, power grid, etc. Most of these graphs have different characteristics such as degree distribution, due to their underlying sampling space, and they evolve with time. Several questions arise in connection with the practical use of these graphs, including how to extract the most "essential ingredients" in these massive and heterogeneous graphs without strong model assumptions? How to efficiently predict links in very large graphs of scale of 10^5 vertices or more? This short paper aims to address these two fundamental problems in dynamic graphs, and uses an ARMA regression model for predicting links based on time series of graph spectra. Preliminary results on synthetic datasets show the promise of the proposed approach.

1 Introduction

Dynamic graph is a special graph that evolves with time. Most of the social networks are instances of dynamic graph. Some instances range from social networks (DBLP co-authorship, Twitter, Enron email dataset) to biological networks (Protein Protein Interaction, Gene Co-Expression). There are several survey papers on link prediction [3,4].

Here we define the dynamic graph and its link prediction problem as follows:

Definition 1.1 (Dynamic Graph)**.** *Denote a graph $G(V, E)$,where V is the vertex set and E the edge set. The set $\{G_t, t = 1, 2, ..., T\}$, is a dynamic graph, where G_t denotes the graph at time moment t.*

Note that in this paper, G_t is an unweighted undirected graph and shares the same vertex set V across the whole time series. This means that $G_t = (V, Et)$.

Definition 1.2 (The link prediction in dynamic graph)**.** *Given a dynamic graph, $\{Gt, t = 1, 2, ..., T\}$, the link prediction problem consists in inferring a set of new edges in this graph, to obtain the graph $G_{T+1} = (V, E_{T+1})$ where E_{T+1} is the set of edges at $T + 1$.*

Spectral graph theory [1] decomposes the graph in the discrete world by solving the eigenvalue problem of the graph Laplacian matrix in the continuous universe, and can extract those "essential

*NIPS 2010 Workshop for Low-rank Methods for Large-scale Machine Learning, Whistle, Canada

ingredients" of a graph so we can understand the graph with more insights. Low Rank Approximation demonstrates its power in this paper to reconstruct the matrix from the linear combination of the outer product of eigenvectors, and recently shows its promise when the scale of the real world graphs are too large which makes a lot of full matrix computations almost intractable.

Definition 1.3 (Graph spectra)**.** *The graph spectra for an undirected unweighted graph is the collection of eigenvalues of the corresponding graph Laplacian matrix. More specifically, the normalized graph Laplacian matrix is defined as:*

$$L = I - D^{-1/2}AD^{-1/2}$$

where D is a diagonal matrix with vertex degrees, A is the adjacency matrix, I is the identity matrix.

Details of this derivations is in [1], which basically is to solve the Rayleigh Quotient problem on Riemann manifold with Laplace-Beltrami operator.

Definition 1.4 (Low Rank Approximation for graph Laplacian)**.** *The low rank $(K \leq N)$ approximation of the graph Laplacian matrix is given by $H_t^K = \sum_{i=1}^{K} \lambda_i \vec{x}_t^i \vec{x}_t^{iT}$*

where $\vec{x}_t^i$ and λ_i denote, respectively, the orthonormal eigenvectors and the corresponding eigenvalues of the graph Laplacian matrix.

In previous literature, lots of approaches focus on various similarities from graph theory to help predict the possible links [3,4,5], and some of them tried to apply statistics to predict the overall properties of the graph [6]. As stated in [5], most algorithms still perform poor with accuracy around 5%, partly due to the wrong model assumptions on the heterogeneous graphs. A more general framework with less model assumption will improve the predictive power in dynamic graphs. Time series analysis seeks statistical models a sequence of successive data points [2], and AutoRegressive Moving Average model is a sophisticated time series model for regressions and predictions on temporal signals.

Definition 1.5 (ARMA model)**.** *Given a time series $X_t, t = 1, 2, ..., T$, The ARMA model ARMA(p, q) refers to the model with p autoregressive terms and q moving average terms, and its defined as:*

$$X_t = c + \epsilon_t + \sum_{i=1}^{p} \phi_i X_{t-i} + \sum_{i=1}^{q} \theta_i \epsilon_{t-i}$$

where constant c and θ_i are model parameters and need to be optimally computed.

However, simply using ARMA model, it is still difficult to predict the graph matrix as a whole directly and more importantly, less insightful about the graph natures. To avoid the difficulties described meaningful features need to be extracted from the graph with the aim to observe and predict their evolution in time. Therefore it is natural to combine the spectral graph theory and low rank approximation with time series to construct a better algorithm for dynamic graph link prediction problem. As the dynamic graph becomes large, the computation complexity emerges. How to efficiently predict links in very large graphs of scale of 10^5 vertices (number of genes or proteins in a model organism) or more (Twitter followers graph, Internet webpage links)? Low rank approximation serves as an efficient tool in decomposing the large graphs, and with scalable complexity [8] in most cases due to the sparsity of the Laplacian matrix.

2 Algorithm

We start with 3 lemmas to prove the Therem 1, which leads to our algorithm.

Lemma 2.1. *The ARMA model is optimally fitted by least squares regression to find the values of the parameters which minimize the error term. More specifically, graph spectra is the optimal least squares regression based prediction from $\{\vec{x}_t^i, t = 1, 2, ..., T\}$*

Remark: To avoid over-fitting, it is generally considered good practice to find the smallest values of p and q which provide an acceptable fit to the data, and ACF and PACF can be used as a good analysis tool [2]. For a proof of this result refer to [2].

Lemma 2.2. • *Let H_t^K denotes the estimate of the graph spectra with K eigenvectors at time t. That is, $H_t^K = \sum_{i=1}^K \lambda_i \vec{x}_t^i \vec{x}_t^{iT}$, then we have $\lim_{K\to N} ||H_t^K - G_t||_F = 0$ where $N = |V|$.*

- *For a connected graph G with N nodes, the rank of the graph Laplacian matrix is N-1.*
- *From the semi-positive definite property of the Laplacian matrix, $\lambda_i \geq 0$*

Lemma 2.3. *Let N be the number of vertices in the graph, and $K \leq N$. Then $\sum_{i=1}^K \lambda_i \leq N$ with equality holding if and only if the graph is connected.*

Theorem 2.4 (Optimality of Graph Spectra Regression)**.** *Given the dynamic graph $\{G_t, t = 1, ..., T\}$ on N vertices, and $K \leq N$ the low rank approximation of the graph Laplacian, H_{T+1}^K is the optimal estimator of the spectrum of the graph G_{T+1}.*

Proof. From Lemma 1, it follows that the set of graph spectra is optimal in the sense of minimizing least square error of the objective function in the ARMA model. Hence we obtain a good estimator of the graph spectra. By Lemma 2, the low-rank approximation of the rank (N-1)-matrix corresponding to Gt can be minimized monotonically when the number of graph spectra increases. Hence we can reconstruct Gt monotonically with sufficient graph spectra. In addition to the facts derived from Lemmas 1 and 2 we need one more piece of information, in order to obtain the optimal estimator for graph G_{T+1}. We observed from empirical data that, when the dynamic graph evolves gradually (few edges are added or removed at each successive time moment) the quantity $\sum_{i=1}^N \frac{|\lambda_{t+1}^i - \lambda_t^i|}{|\lambda_{t+1}|}$ is about 6%. □

Graph Spectra Regression Link Prediction Algorithm

- For each time point t, compute the normalized Laplacian matrix for G_t;
- For each G_t , solve the generalized eigenvalue problem for its Laplacian to obtain a set of K eigenvectors. K is an algorithmic parameter which controls the prediction accuracy.
- For each cell element $\vec{x}_t^i(p)$ in each eigenvector, collect the successive time series values $\{\vec{x}_t^i(p), t = 1, 2, ..., T\}$, and compute the ARMA model.
- From this model, we can predict a set of K eigenvectors $\vec{x}_{t+1}^i$.
- Use the eigenvectors λ_t as an estimator of λ_{t+1}.
- Use the low rank approximation to reconstruct the graph Laplacian of G_{t+1}, which is the predicted graph. Link prediction can be obtained by comparing the edges in this graph with G_t.

Worth mentioning, the eigenvectors corresponding to smallest eigenvalues are optimal in solving the Raleigh Quotient problem, but its contribution to the low rank approximation in reconstructing the predicted graph Laplacian is very limited. Therefore, smallest eigenvectors reflect the graph cut property, which encode the cluster structures; largest eigenvectors, on the other hand, preserve the neighborhood structure and contribute more to the reconstruction of the graph. Finding a hierarchical framework to utilize both ends can be a promising future work.

3 Experiment

We synthesize a dynamic graph of 100 vertices that consists of 10 graphs from 10 time points. The degree of node 1 is increasing along the time, simulating an active node that is very likely to attach to other nodes. The node 1 is Incrementally attached to node 51, 52, ... 60, forming a sequence of 10 graphs as in Figure 1.

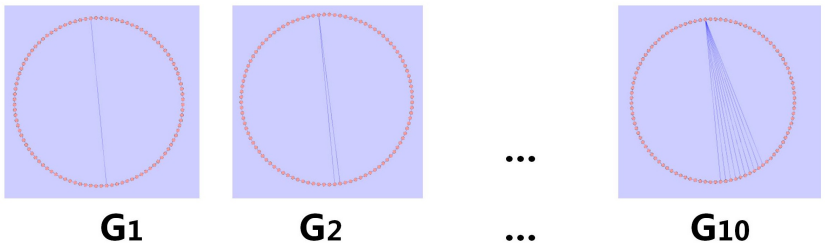
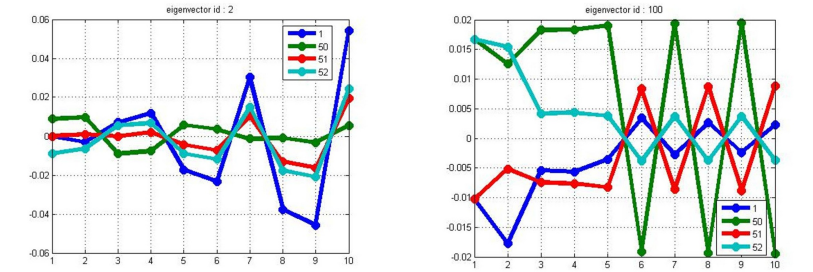


Figure 1, synthetic dynamic graphs



2nd smallest Eigenvector
Node 1 ARMA loss: 0.0005
A(q) = 1 + 0.3344 q^-1 + 0.6572 q^-2

largest Eigenvector
Node 1 ARMA loss: 1.07858e-005
A(q) = 1 - 0.2767 q^-1 - 0.4177 q^-2

Figure 2, time series of 4 nodes for 2nd smallest and the largest eigenvector.

Interestingly we found that in most of the eigenvectors, node 1 has the most tendencies to be increasing, as Figure 2 blue curve. For the node 50 (the green curve in Figure 2) as comparison, its tendencies to be monotonically rise or fall is not identifiable.

We report the ARMA model loss and its model A(q) for node 1, and as shows in Figure 2, the ARMA fitting results are with high confidence in order 2 ARMA model.

References

[1] Chung, F.R.K., Spectral Graph Theory, American Mathematical Society,1997.

[1] George, B., Time Series Analysis: Forecasting & Control, 3rd Ed, Pearson Education, 1994.

[2] Lu, Linyuan; Zhou, Tao, Link Prediction in Complex Networks: A Survey, 2010arXiv1010.0725L, 2010.

[3] Lise Getoor, Christopher P. Diehl, Link Mining: A Survey, SIGKDD Explorations, Volume 7, Issue 2

[4] David Liben-Nowell, Jon Kleinber, The link-prediction problem for social networks, Journal of the American Society for Information Science and Technology, Volume 58, Issue 7, pages 1019–1031, May 2007.

[5] Zan Huang, Link Prediction Based on Graph Topology: The Predictive Value of Generalized Clustering Coefficient, ACM LinkKDD 2006.

[6] Nalini Ravishanker, Dipak Dey , A first course in linear model theory, Chapman and Hall/CRC, 2002.

[7] J Ye, Generalized Low Rank Approximations of Matrices, Journal of Machine Learning, 2005, Springer.