SYNTHETIC TEMPORAL GRAPH GENERATION

Sumit Purohit[1], *Lawrence B. Holder*[2], *George Chin*[1]

SIAM Workshop on Network Science 2018
July 12-13 · Portland

Abstract

Generating a synthetic graph that is similar to a given real-world graph is a critical requirement for privacy preservation and benchmarking purposes. Various generative models attempt to generate static graphs similar to real-world graphs. However, generation of temporal graphs is still an open research area. We present a temporal-motif based approach to generate synthetic temporal graph datasets including the core algorithm, and results from two real-world use cases.

Introduction

Graphs are a natural and flexible representation of a set of entities and the relationships among them. A static graph represents a set of objects and a set of pairwise relations between them. A temporal graph is a generalization of a static graph which changes with time. Time can also be modeled as a vertex or edge label, which makes temporal graphs a special case of attributed graphs. Incorporating time into the static graphs has given rise to a new set of challenging and important problems that can not be modeled as a static-graph problem [6]. Many domains such as social networks, communication, transportation, sensor networks, co-authorship networks, and procurements can be naturally modeled as temporal graphs.

Many graph generative models are studied and developed to generate synthetic graphs. Random Model [4] and Preferential Attachment Model [2] are classic graph generative models. The Chung-Lu model provides a random model to generate power law graphs [1] using an input degree distribution. Recently Leskovec and Foloutsos [5]presented the Kronecker model based on Kronecker matrix multiplication to generate syntactic graphs that replicate multiple graph properties. All such models attempt to satisfy some global graph properties, but do not guarantee the preservation of localized structural properties.

[1]Pacific Northwest National Laboratory, Richland, WA, 99352
[2]Washington State University, Pulllman, WA, 99164, USA

This research presents a graph generative model that preserves local temporal structures while generating synthetic graphs. It defines some easy to compute temporal atomic motifs which are used to define any real-world graph. The core hypothesis of this research is that preserving local temporal-motifs is sufficient to generate synthetic graphs that also exhibit similar global graph properties of the corresponding real-world graph.

Structural Temporal Modeling

We define Structure Temporal Modeling (STM) as a process of identifying temporal-motifs in the real-world graph. We define some easy to compute atomic-motifs such those shown in Figure 1 which can characterize any given real-world graph. We guarantee that the motifs are found in mutually exclusive fashion and we do not find overlapping motifs. We define *vertex-birth-time* of a vertex as the earliest arrival time of temporal edges associated with this vertex. We define *motif-birth-time* as the earliest time at which any edge of that motif has arrived. Using these two definitions we compute the information content of a motif as the number of new and old vertices associated with the motif. This leads to multiple *temporal-atomic-motifs* for a given *atomic-motif*. For example, in Figure 1 a triangle atomic-motif is expanded to 4 temporal-atomic-motifs where 0,1,2, or 3 vertices are new (or re-used). The six atomic-motifs in Figure 1 can generate up to 20 temporal-atomic-motifs.

For each temporal-atomic-motif we also compute its *formation-time* which is the total time taken by the motif to fully form. At the same time, we also compute *average-arrival-delay* in generating each edge of the motif.

Distribution of such temporal-atomic-motifs is computed for a given real-world graph. Motif *arrival-rates* are computed by normalizing the distribution over the entire duration of the input graph. This normalized distribution is used to generate its synthetic version and the same distribution is also computed for the synthetic graph. Variation in these two distributions is used as a metric to compare quality of the synthetic graph.

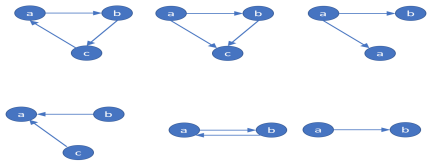

Figure 1: Atomic Temporal Motifs

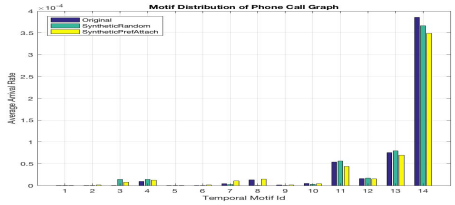

Figure 2: Synthetic Communication Network

The generator component of the STM uses the distribution to iteratively generate all the temporal motifs using arrival rates as *generation probabilities*. STM uses the information content of the motifs to decide whether to create new nodes or reuse existing nodes in the graph at a given point of time. STM also uses *formation-time* and *average-arrival-rate* to delay the formation of the temporal-motif.

Experiments

We have developed a scalable framework using Apache Spark [7] and GraphFrames [3] to compute the distribution of temporal-atomic-motifs. We have also developed a graph generator using Python (https://github.com/lbholder/graph-stream-generator) that takes the distribution as an input and generates a synthetic graph. We present results from two domains: social networks and communication networks. We were able to model one million edge graphs successfully.

Figure 2 shows the temporal motif distribution of real and synthetic snapshots of the PNNL internal communication network where each edge represents a phone communication between two persons. Similarly, Figure 3 shows the temporal motif distribution of real and synthetic Twitter graphs generated using the public API, where each edge represents a Twitter mention by source to destination. We experimented with two variations of the synthetic graph generation, random node selection and preferential node selection where a reused node is selected based on its degree. As shown in Figures 3 and 2, STM generates synthetic graphs similar to corresponding real-world graphs. It is also quantitatively evident from the very low absolute mean difference value of the motif probabilities as shown in Table below.

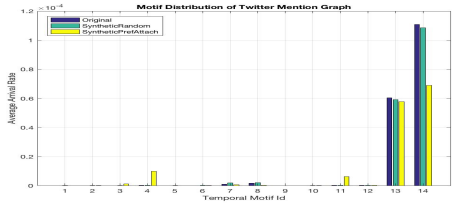

Figure 3: Synthetic Social Network

	Random	Degree
Social Network	3.7398e-07	4.5396e-06
Communication	4.4522e-06	5.3076e-06

Future Work

Future work will model multi-type graphs that increase the number of candidate temporal motifs. We will address this challenge.

Acknowledgment

We thank the DARPA Modeling Adversarial Activity program for funding this project under contracts HR0011728117, HR001178235, and HR0011729374. The associated PNNL project number is 69986. This work is also supported by the National Science Foundation under Grant No. 1646640.

References

[1] W. Aiello, F. Chung, and L. Lu. A random graph model for massive graphs. In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 171–180. Acm, 2000.

[2] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[3] A. Dave, A. Jindal, L. E. Li, R. Xin, J. Gonzalez, and M. Zaharia. Graphframes: an integrated api for mixing graph and relational queries. In *Proceedings of the Fourth International Workshop on Graph Data Management Experiences and Systems*, page 2. ACM, 2016.

[4] P. Erdos. On random graphs. *Publicationes mathematicae*, 6:290–297, 1959.

[5] J. Leskovec, D. Chakrabarti, J. Kleinberg, and C. Faloutsos. Realistic, mathematically tractable graph generation and evolution, using kronecker multiplication. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 133–145. Springer, 2005.

[6] O. Michail. An introduction to temporal graphs: An algorithmic perspective. *Internet Mathematics*, 12(4):239–280, 2016.

[7] A. Spark. Apache spark: Lightning-fast cluster computing. *URL http://spark. apache. org*, 2016.