Sampling Subgraph Density in Large-Scale Information Networks (DRAFT)

Zongmian Li **Sylvain Truong** **Jan Ramon** **Juan Alvarado**

Abstract

In large information networks (e.g. social networks, biological networks), the density of subgraph patterns (also called motifs or graphlets) can provide information on how groups of objects interact. For example, in a social network, if X and Y, Y and Z are friends, then the probability that X and Z are friends may be higher than for a randomly selected pair of persons. The notion of "clustering coefficient" has been widely studied to measure the density of triangles in a graph (as compared to what one expects from the edge density).

This project is situated in the attempt to generalize over the notion of clustering coefficient and study the behavior of graphs satisfying a number of subgraph density constraints (e.g. we know the edge density and triangle density) but which for the rest are supposed to be as random as possible (i.e. the entropy is as high as possible). This project is concerned with developing the theory and implementation of an algorithm to sample (i.e. randomly draw) large graphs satisfying a set of subgraph density constraints. The main idea is to develop a Metropolis-Hasting style algorithm.

We may know how many edges and how many triangles a graph has, and could be interested in knowing how many 4-cliques we could expect. Estimating the expected amount of 4-cliques would be possible using the planned sampling algorithm. In the longer run we are also interested in applications in anomaly detection.

This class project involves non-trivial components of algorithm design, implementation and empirical evaluation.

1 Introduction

This project involves a few notions of graph and sampling theory, which we first introduce here, before making the problem statement.

1.1 Random graphs

The theory of random graphs focuses on the study of probabilistic models on graphs. One could say that this theory lies somewhere between graph theory and probability theory. One of the main interest of the theory of random graphs is to do approximate inference: from a given probability distribution over a set of graphs, which possibly share a certain number of constraints, we can design techniques to sample graphs from this distribution. Further applications could involve, for example, empirical studies on graphs that have certain desired properties.

The most fundamental and widely studied random graph model is the Erdős-Renyi (ER) model. There are two closely related ER models:

- The ER-m model $G(n, m)$ ($n, m \in \mathbb{N}$), which represents the set of graphs with n nodes and m edges. Given n and m, a graph G in $G(n, m)$ is chosen at random with uniform probability over $G(n, m)$, i.e., $p(G) = 1/C_{C_n^2}^m$.

- The ER-p model $G(n, p)$ ($n \in \mathbb{N}$, $p \in [0, 1]$). A graph is constructed by connecting each pair of vertices at random with probability p. Therefore, the probability of choosing a graph G which has n vertices and m edges is $p(G) = p^m(1-p)^{C_n^2 - m}$.

Sampling graphs in the ER framework is pretty straightforward. Now let us take a look at further properties that characterize a graph. The focus of the project would then be about formulating sampling strategies over graphs that possibly share more complex properties.

1.2 Subgraph densities

Densities on graphs are defined for graphs motifs (or graphlets). For a given motif, they are simply defined as the number of motifs that are present, over the maximum possible total number of motifs. For instance, for a graph featuring n vertices and m edges, the edge density is defined as follows:

$$p_2 = \frac{|m|}{|n|(|n|-1)}, \quad \text{for directed graphs,}$$
$$p_2 = \frac{2|m|}{|n|(|n|-1)}, \quad \text{for undirected graphs.}$$

Similarly, one can further define the triangle density p_3, the 4-clique density p_4 and so on.

1.3 Problem statement

The goal of this project is to propose a graph sampling strategy under subgraphs density constraints. The constraints relative to the number of vertices n, the edge density p_2 and the triangle density p_3 constraints will firstly be considered.

The proposed approach will be an elaboration of the Metropolis-Hastings algorithm (as shown in Algorithm 1). Given constraints on n and p_i ($i \geq 2$), the first step is to define a probability distribution over n-vertex graphs which attribute larger probabilities to graphs satisfying the constraints. For instance, if we are given n and a p_2 constraint, a good posterior probability candidate F would be

$$\forall G \in G(n),\ F(G) \propto \exp(-M(p_2(G) - p_2)^2), \quad \text{with } M \in \mathbb{R}_+, \tag{1}$$

where $p_2(G)$ is the number of edges in the instance graph G. Note that to implement the Metropolis-Hastings algorithm, knowing the probability distribution up to a multiplicative constant is enough. Therefore, defining probabilities in the form of Equation (1) is sufficient for our approach.

For a large M, the probability distribution will become very concentrated around graphs satisfying the constraints, thus resulting in a very precise sampling algorithm, but at the expense of possibly long burn-in periods in the Metropolis-Hastings algorithm. Therefore, the project also covers an empirical study for parameter tuning and a precision-speed trade-off evaluation.

Another challenge of the project lies in the design of a transition matrix. Although the Metropolis-Hastings is proved to converge for any transition matrix, defining a matrix which is likely to favor transitions to matrices abiding by the constraints could favor the burn-in period. A sensitivity analysis could be led, in regard of this consideration.

Algorithm 1 The Metropolis-Hastings algorithm

Assume we have a probability distribution π over state space $\mathcal{X}$, a proposed transition matrix $T(\cdot, x), \forall x \in \mathcal{X}$

for $0 \leq t \leq T_{max}$ **do**

Given x^t, choose x^{t+1} in the following manner

Draw $z \sim T(\cdot, x^t)$

$q \leftarrow \min(1, \frac{\pi(z)T(x^t|z)}{\pi(x^t)T(z|x^t)})$

With probability q, accept: $x^{t+1} \leftarrow z$

Otherwise, $x^{t+1} \leftarrow x^t$

end for

After a certain number of iterations (*burn-in* period), the sequence (x^t) converges to a Markov chain of stationary distribution π.