Estimating Graph Parameters using Graph Grammars

Sourav Mukherjee[1] and Tim Oates[2]

[1] Department of Computer Science,
University of Maryland, Baltimore County, USA
Phone: +1-410-455-8790
sourav1@umbc.edu
[2] Department of Computer Science,
University of Maryland, Baltimore County, USA
oates@cs.umbc.edu

Stochastic graph grammars are probabilistic models suitable for modeling relational data, complex organic molecules, social networks, and various other data distributions [1]. In this paper, we demonstrate that such grammars can be used to reveal useful information about the underlying distribution. In particular, we demonstrate techniques for estimating the expected number of nodes, the expected number of edges, and the expected average node degree, in a graph sampled from the distribution. These estimation techniques use the underlying grammar, and hence do not require sampling. Experimental results indicate that our estimation techniques are reasonably accurate.

We use the notation $G = (V, E)$ to refer to a graph with V being the set of vertices, and E being the set of edges.

Definition 1. *Let $G = (V, E)$ be a graph. A* hyperedge *is an ordered subset of its vertices V. Alternatively, a hyperedge of degree n can be thought of as a mapping $H : \{1, 2, ..., n\} \rightarrow V$. A* hypergraph *is a graph that can, in addition to edges, also have hyperedges.*

Definition 2. *A* (hyperedge replacement) stochastic context-free graph grammar *(SCFGG) is defined as a tuple (S, N, T, P, p) where:*

- *N is the set of non-terminal symbols,*
- *T is the set of terminal symbols, disjoint from N,*
- *$S \in N$ is a special non-terminal called the start symbol,*
- *P is a set of productions,*
- *p is a probability function defined on the set of productions, such that the sum of the probabilities of all productions with the same left hand side equals 1.*

In a hyperedge replacement SCFGG, terminals are used to denote graphs without hyperedges, while non-terminals are used to label hyperedges. A production is an ordered pair (H, α), written as $H \rightarrow \alpha$, where H is a non-terminal and α is a hypergraph.

A SCFGG can be viewed as a generative model: we start with the start symbol S, and at each step, we replace any non-terminal H with a graph α such that there is a production $H \rightarrow \alpha$. This process is continued until we arrive at a graph that has no non-terminal symbols. When a hyperedge H in a graph G is replaced using the production $H \rightarrow \alpha$, G is called the host-graph, and α is called the subgraph.

We now present techniques for estimating the expected number of vertices and nodes in a graph sampled from a distribution, given the grammar for the distribution. We define the following notation, which will aid the subsequent discussion. For any non-terminal Z, let us assume that there are N_Z production rules with Z on the left hand side, with probabilities $p_{Z,1}, p_{Z,2}, ..., p_{Z,N_Z}$, satisfying $\sum_{j=1}^{N_Z} p_{Z,j} = 1$. Let the j^{th} such production be of the form $Z \rightarrow \alpha_j$ where α_j is a graph with $v_{Z,j}$ vertices, $a_{Z,j}$ edges and $h_{Z,j}$ hyper-edges, labeled $Z_{j,1}, Z_{j,2}, ..., Z_{j,h_{Z,j}}$. Note that these non-terminals do not have to be all distinct; they may even be the same as Z. Finally, let D_Z denote the degree of the non-terminal Z.

For any non-terminal Z, let n_Z represent the expected number of nodes in any graph obtained by expanding Z. Then the equation for n_Z is given by:

$$n_Z = \sum_{j=1}^{N_Z} p_{Z,j}(v_{Z,j} + \sum_{k=1}^{h_{Z,j}} (n_{Z_{j,k}} - D_{Z_{j,k}})) \tag{1}$$

Thus we see that for each non-terminal Z in the grammar, we will have a single linear equation, leading to a system to linear equations with the same number of equations as the number of non-terminals.

We now develop a system of linear equations for estimating the expected number of edges e_Z in a graph, obtained from any non-terminal Z in the grammar. The problem of estimating the expected number of edges is different from that of estimating the expected number of nodes, in that unlike nodes, edges are not glued together when a subgraph is embedded inside a host-graph. The equation for e_Z is given by:

$$e_Z = \sum_{j=1}^{N_Z} p_{Z,j}(a_{Z,j} + \sum_{k=1}^{h_{Z,j}} e_{Z_{j,k}}) \tag{2}$$

Once again, we see that for each non-terminal Z in the grammar, we will have a single linear equation, leading to a system of linear-equations with the same number of equations as the number of non-terminals.

Next we present two techniques, the *Naïve Degree Estimator* and the *Linear Degree Estimator* for estimating the average node degree of a graph generated from a given grammar. The average degree $\bar{d}$ of a node in a graph $G = (V, E)$ is defined as $\bar{d} = \frac{1}{|V|} \sum_{v \in V} d(v)$.We also know that $\bar{d} = \frac{2|E|}{|V|}$. We will refer to this result as the Handshaking Lemma [2].

Given a non-terminal Z, let $\bar{d}_Z$ denote the expected value of the average degree of a node, of any graph obtained from Z. Then, we can estimate $\bar{d}_Z$ as:

$$\bar{d}_Z \approx \frac{2e_Z}{n_Z} \tag{3}$$

Of course, Equation 3 is only an approximate estimate, because the number of nodes and the number of edges are not, in general, independent. We now present a more accurate estimator.

Let, for a non-terminal Z, $\bar{d}_Z$ indicate the expected average node degree of any graph derived from the non-terminal symbol Z. Recall that the average is computed over all nodes in a graph, and the expectation is computed over the distribution of the graphs. Then, the expected number of nodes in the graph α_j is given by

$$n_{Z,j} = \sum_{k=1}^{h_{Z,j}} (n_{Z_{j,k}} - D_{Z_{j,k}}) + v_{Z,j} \tag{4}$$

Let us number the vertices in α_j as $1, 2, ..., v_{Z,j}$ and let for vertex $l (1 \leq l \leq v_{Z,j})$, a_l be the number of terminal edges incident on that vertex. Then the expression for the expected average number of nodes is given by:

$$\bar{d}_Z - \sum_{j=1}^{N_Z} \sum_{k=1}^{h_{Z,j}} \frac{p_{Z,j}}{n_{Z,j}} \bar{d}_{Z_{j,k}} = \sum_{j=1}^{N_z} \sum_{l=1}^{v_{Z,j}} \frac{p_{Z,j}}{n_{Z,j}} a_l \tag{5}$$

Thus, we get a linear equation for every non-terminal Z in the grammar. By solving this linear system, we can arrive at an estimate of the expected average node degree.

Graph grammars are useful probabilistic models for distributions over graphs because they are compact, hierarchical, and amenable to interpretation by domain experts. However, in this paper, we have demonstrated that the utility of graph grammars goes beyond elucidation of structure and generation of samples. We have presented grammar-based techniques to estimate the expected number of nodes, the expected number of edges, and the expected average node degree in a graph generated by the grammar. We have also presented a characterization of grammars that can produce graphs that are not connected. Future directions include exploring the characterization of grammars the generate planar graphs, and applying these results to real-life domains such as relational databases, organic molecules, and social networks.

References

1. Grzegorz Rozenberg, editor. *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations.* World Scientific, 1997.
2. D. B. West. *Introduction to Graph Theory (2nd Edition).* (Prenctice Hall, Upper Saddle River), 2001.