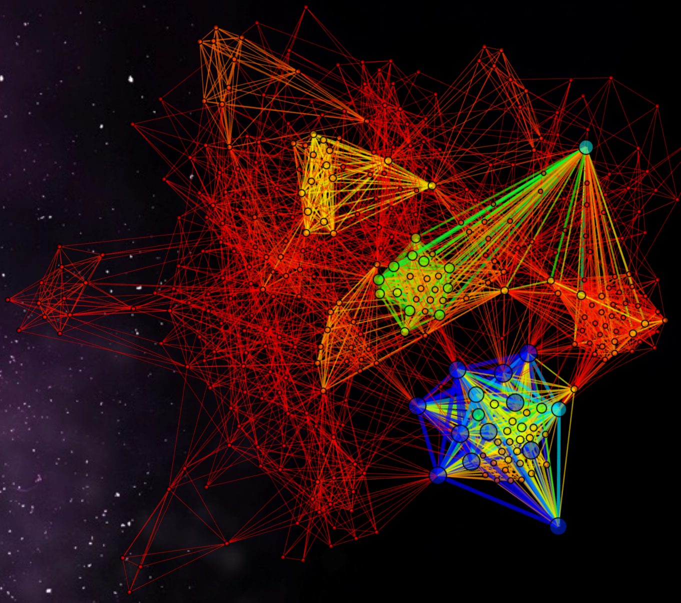
Fast and Accurate Estimation of Typed Graphlets

Ryan A. Rossi[1] | Anup Rao[1], Tung Mai[1], Nesreen K. Ahmed[2]

[1] Adobe Research

[2] Intel Labs



Problem

(a) Typed 4-cycles with 2 types (b) Typed 4-cliques with 2 types

- Typed graphlets = small typed (labeled) induced sub-graphs
- Generalization of graphlets to labeled and heterogeneous networks
- Useful for many applications including clustering, link prediction, network alignment, etc.

Exact Problem. Given a graph G with L types, the global typed graphlet counting problem is to find the set of all typed graphlets that occur in G along with their corresponding frequencies

- Depending on the application constraints, speed may be more important than accuracy
  - e.g., applications requiring real-time response rates such as online recommendation, online advertisements, among many others

QUESTION: Can we instead obtain fast estimates with provable error guarantees?

Problem

(a) Typed 4-cycles with 2 types (b) Typed 4-cliques with 2 types

- Typed graphlets = small typed (labeled) induced sub-graphs
- Generalization of graphlets to labeled and heterogeneous networks
- Useful for many applications including clustering, link prediction, network alignment, etc.
- Depending on the application constraints, speed may be more important than accuracy

Our Problem. Given a graph G with L types, the typed graphlet estimation problem is to accurately estimate the counts of all typed graphlets that occur in G while achieving orders of magnitude speedup compared to exact algorithms.

Problem

(a) Typed 4-cycles with 2 types (b) Typed 4-cliques with 2 types

- Typed graphlets = small typed (labeled) induced sub-graphs
- Generalization of graphlets to labeled and heterogeneous networks
- Useful for many applications including clustering, link prediction, network alignment, etc.
- Depending on the application constraints, speed may be more important than accuracy

Our Problem. Given a graph G with L types, the typed graphlet estimation problem is to accurately estimate the counts of all typed graphlets that occur in G while achieving orders of magnitude speedup compared to exact algorithms.

Balance tradeoffs: tiny decrease in accuracy for significant improvement in runtime (100-1000x speedup)

Framework for Typed Graphlet Estimation

Introduce two general classes of typed graphlet estimation methods:

1. Typed *Edge* Sampling (TES) & Estimation

2. Typed *Path* Sampling (TPS) & Estimation

Typed Edge Sampling (TES)

(a) Typed 4-cycles with 2 types (b) Typed 4-cliques with 2 types

- Typed Edge Sampling $J \subseteq E$

- Estimation $\mathbf{X}_H = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots \end{bmatrix} \in \mathbb{R}^{|J| \times |\mathcal{H}|}$

(typed graphlet counts that occur at each sampled edge in J for a specific induced subgraph H (e.g., 4-clique) and $\mathcal{H}$ is the set of typed graphlets of H)

$$\widehat{\mathbf{x}}_H = \left(\frac{|J|}{|E|} \right)^{-1} \frac{\mathbf{e}^T \mathbf{X}_H}{|E(H)|}$$

Typed Path Sampling (TPS)

- Sampling

DEFINITION 1 (TYPED WEDGES). *Given an edge $(i,j) \in E$ with types ϕ_i and ϕ_j, the (i,j)-entry of the typed wedge matrix with types t and t' is:*

$$\Lambda_e^{tt'} = \Lambda_{ij}^{tt'} = \begin{cases} (d_i^t - 1)(d_j^{t'} - 1) & \text{if } t = \phi_j \wedge t' = \phi_i \\ (d_i^t - 1)d_j^{t'} & \text{if } t = \phi_j \wedge t' \neq \phi_i \\ d_i^t(d_j^{t'} - 1) & \text{if } t \neq \phi_j \wedge t' = \phi_i \\ d_i^t d_j^{t'} & \text{otherwise} \end{cases} \tag{3}$$

Algorithm 1 Typed Path Sample

Output: the four sampled nodes (i', i, j, j') with types (t_1, t, t', t_2) that form a typed 4-path with edges $\{(i', i), (i, j), (j, j')\}$

Compute $\Lambda_e^{tt'}$ (Eq. 3) for all *typed* edges and set

$$p_e^{tt'} = \Lambda_e^{tt'} / W^{tt'}$$

Select $e = (i, j)$ of type $t_e = (t, t')$ with probability $p_e^{tt'}$

Select $i' \in \Gamma_i^{t_1}$ with type t_1 uniformly at random s.t. $i' \neq j$ if $\phi_j = t_1$.

Select $j' \in \Gamma_j^{t_2}$ with type t_2 uniformly at random s.t. $j' \neq i$ if $\phi_i = t_2$.

Typed Path Sampling (TPS)

- Estimation

$C_{i,\mathbf{t}}$ = # of occurrences of the i-th typed induced subgraph with types **t**

Algorithm 2 Estimation via Typed Paths

Input: graph G, # samples k

Output: estimated counts for all typed 4-node graphlets

1 Obtain k samples (sets of vertices) by running Alg. 1 k times where S_j denotes the j-th set of vertices.

2 **parallel for** $j = 1, \ldots, k$ **do**

3 Determine subgraph induced by S_j (and type vector $\mathbf{t}$)

4 If this is the i-th graphlet with types $\mathbf{t}$, increment $F_{i,\mathbf{t}}^{tt'}$

5 Increment $k^{tt'}$ where t, t' are the other two node types

6 **for** $i \in [2, 6]$ and type vector $\mathbf{t}$ **do**

7 **for all** $t, t' \in \{1, \ldots, L\}$ **do** Set $\widehat{C}_{i,\mathbf{t}} = \widehat{C}_{i,\mathbf{t}} + (F_{i,\mathbf{t}}^{tt'}/k^{tt'}) \cdot \frac{W^{tt'}}{A_{2,i}}$

8 Set $\widehat{C}_{i,\mathbf{t}} = \widehat{C}_{i,\mathbf{t}}/2$

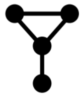
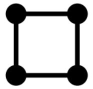
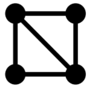
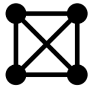
9 Set $\widehat{C}_{1,\mathbf{t}} = N_{1,\mathbf{t}} - \widehat{C}_{3,\mathbf{t}} - 2\widehat{C}_{5,\mathbf{t}} - 4\widehat{C}_{6,\mathbf{t}}$, $\forall \mathbf{t}$ s.t. $N_{1,\mathbf{t}}$ is computed via Eq. 4

A_{ij} = # of distinct copies of the i-th typed subgraph in the j-th subgraph

$N_{i,\mathbf{t}}$ = count of the i-th typed non-induced subgraph with types **t**

Results

Mean relative error of typed graphlet estimates.
We set k = 50000 and perform 100 runs.

Data	Methods					
fb-political	TES	0.012	0.033	0.036	0.036	0.070
	TPS	**0.002**	**0.021**	**0.010**	**0.024**	**0.034**
yahoo-msg	TES	0.774	0.867	1.233	2.784	0.430
	TPS	**0.001**	**0.046**	**0.003**	**0.270**	**0.025**
web-polblogs	TES	0.010	0.083	0.011	0.100	0.164
	TPS	**0.002**	**0.006**	**0.005**	**0.007**	**0.008**
soc-wiki-elec	TES	0.684	1.160	0.788	1.353	1.552
	TPS	**0.002**	**0.003**	**0.003**	**0.005**	**0.008**
soc-digg	TES	0.460	0.412	0.890	0.713	1.474
	TPS	$<10^{-3}$	**0.004**	**0.003**	**0.006**	**0.011**

Variance of estimates

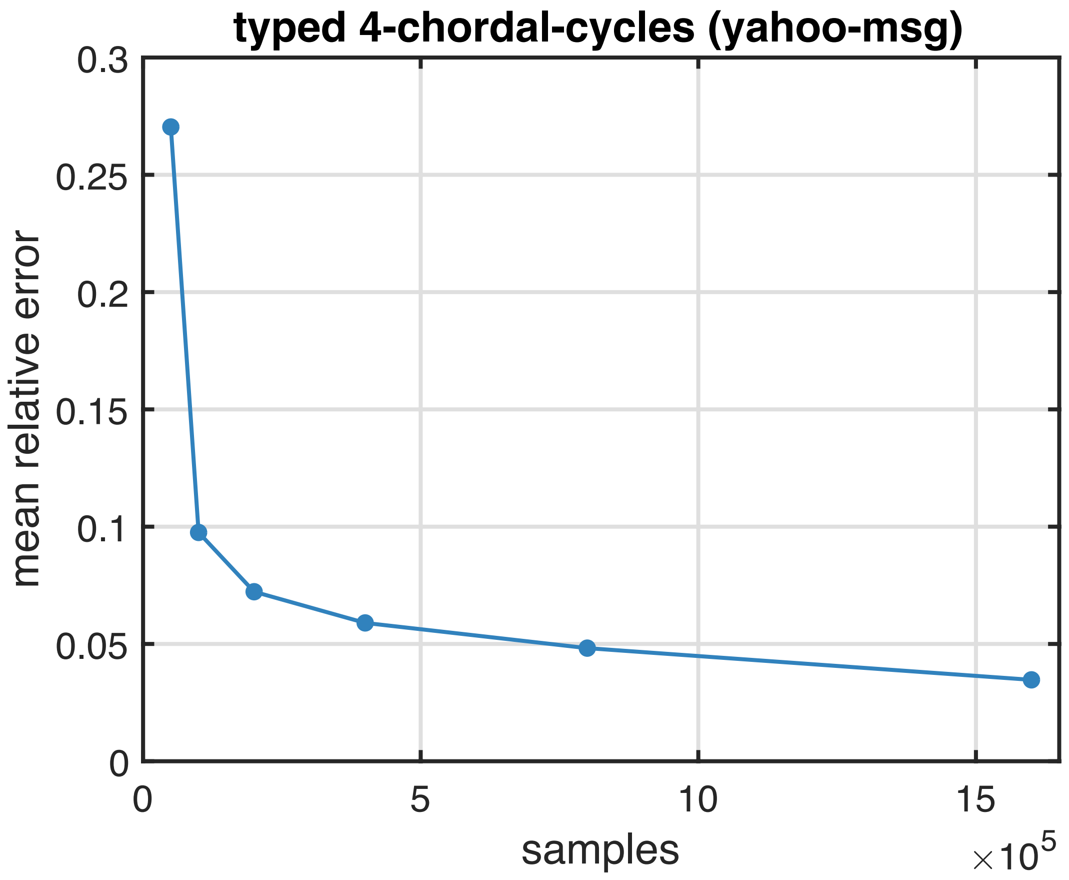
- The standard deviation/variance of TPS to be about an order of magnitude smaller than TES

Table 2: Typed graphlet estimates and relative error using $k = 50000$ (typed 4-cliques).

			$\widehat{C}$		$\frac{\lvert C-\widehat{C}\rvert}{C}$		Std	
graph	**types**	C	**TES**	**TPS**	**TES**	**TPS**	**TES**	**TPS**
fb-political	1111	8.14K	7.35K	8.37K	0.0973	0.0288	5K	507
	2111	7.64K	8.13K	7.86K	0.0634	0.0285	3.4K	545
	2211	6.27K	6.85K	6.50K	0.0924	0.0371	2.6K	448
	2221	6.12K	6.55K	6.29K	0.0701	0.0281	2.3K	455
	2222	4.46K	4.59K	4.68K	0.0274	0.0483	2.4K	462

Convergence Results

- Convergence of estimates
- Increasing the sample size k decreases error of TPS
- Error decreases towards zero as the sample size increases



Summary of Contributions

- Proposed estimation framework for Typed Graphlets
- Described two generic estimators
  - Typed Edge Sampling & Estimation (TES)
  - Typed Path Sampling & Estimation (TPS)
- TPS achieves better accuracy (lower rel. error) & lower variance than TES
- Convergence of estimates (error decreases towards zero as sample size increases)
- Speedup is significant taking less than a second for all graphs (100+ times faster than exact alg.)

Thanks for listening!

Appendix