ABRA: Approximating Betweenness Centrality in Static and Dynamic Graphs with Rademacher Averages

(CoRR, Febuary 2016)

Matteo Riondata and Eli Upfal

Table of Contents

Table of Contents

Is that node important?

Let $G = (V, E)$ be a graph with $|V| = n$ nodes and $|E| = m$ edges.

QUESTION: can we find the most important node in G?

Definition

(Centrality Measure) Function $f : V \mapsto \mathbb{R}^+$ expressing the importance of a node.

MOTIVATION: Find **relevant** web-pages on the web, **influential** participants in a social network, etc.

EXAMPLES: degree, PageRank, closeness, betweeness, etc.

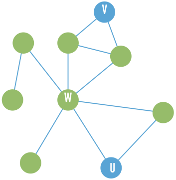
Betweeness Centrality(BC)

Definition

Given a graph $G = (V, E)$, the Betweenness Centrality (BC) of a vertex $w \in V$ is defined as

$$\mathrm{b}(w) = \frac{1}{|V|(|V|-1)} \sum_{\substack{(u,v) \in V \times V \\ u \neq v}} \frac{\sigma_{uv}(w)}{\sigma_{uv}}$$

For any ordered pair (u, v) of different nodes $u \neq v$, let $\mathcal{S}_{uv}$ be the set of Shortest Paths (SPs) from u to v, and let $\sigma_{uv} = |\mathcal{S}_{uv}|$. Denote $\sigma_{uv}(w)$ as the number of SPs from u to v that goes through w.



Rademacher Average

Let $\mathcal{F}$ be a family of functions from $\mathcal{D}$ to $[0,1]$, and let $\mathcal{S} = \{c_1, \ldots, c_\ell\}$ be ℓ i.i.d samples from $\mathcal{D}$. For each $f \in \mathcal{F}$, the true sample and the sample average of f on a sample $\mathcal{S}$ are

$$\mathrm{m}_{\mathcal{D}}(f)\frac{1}{|\mathcal{D}|}\sum_{c \in \mathcal{D}}^{\ell} f(c) \text{ and } \mathrm{m}_{\mathcal{S}}(f) = \frac{1}{\ell}\sum_{i=1}^{\ell} f(c_i)$$

Theorem

(Bounding Maximum Deviation) Let $\delta \in (0,1)$ and let $\mathcal{S}$ be a collection of ℓ i.i.d samples from $\mathcal{D}$. Then, with probability at least $1-\delta$,

$$\sup_{f \in \mathcal{F}} |m_{\mathcal{S}}(f) - m_{\mathcal{D}}(f)| \leq 2\mathrm{R}(\mathcal{F}, \mathcal{S}) + 3\sqrt{\frac{\ln(2/\delta)}{2\ell}}$$

Where

$$\mathrm{R}(\mathcal{F}, \mathcal{S}) = \mathbb{E}_\sigma\left[\sup_{f \in \mathcal{F}} \frac{1}{\ell}\sum_{i=1}^{\ell} \sigma_i f(c_i)\right]$$

Table of Contents

Contribution of this paper

- Progressive sampling based BC approximation within ε additive factor
- First BC approximation algorithm to estimate BC without depending on any global property of the graph
  - Related work: RK algorithm [Riandato and Karnopoulis 2016] depends on Vertex diameter of the graph

Definition

Given $\varepsilon, \delta \in (0, 1)$, *an* (ε, δ)*-approximation to* B *is a collection* $\tilde{B} = \{\tilde{b}(w), w \in V\}$ *such that*

$$\Pr(\forall w \in v : |\tilde{b}(w) - \mathrm{b}(w)| \leq \varepsilon) \geq 1 - \delta$$

Progressive Sampling Algorithm

input:

Graph **G=(V,E)**,
accuracy parameter $\varepsilon \in (0, 1)$,
confidence parameter $\delta \in (0, 1)$

output:

Set $\widetilde{B}$ of bc approximations for all nodes in V

$S_0 \leftarrow 0$
$\boxed{S_1 \leftarrow \text{initial\_sample\_size}()}$
i $\leftarrow$ 0

D$\leftarrow \{(u, v) \in VXV, u \neq v\}$
while $\boxed{\textbf{stopping condition}}$ **not satisfied:**
for $l \leftarrow 1$ to $S_i - S_{i-1}$
(u,v)$\leftarrow$ uniform_random_sample(D)
compute_SPs(u, v)
calculate $\widetilde{B}$ from sample S_i
i $\leftarrow$ i+1
$\boxed{S_i \leftarrow \text{next\_sample\_size}()}$

Random sampling to approximate betweeness

Works	Sample Space	Sample Size for (ε, δ)-approximation *	Analysis Techniques
[Jacob et al. 2005], [Brandes and Pich 2007] [Hayashi et al. 2015]	nodes	$O\left(\frac{1}{\varepsilon^2}\left(\ln \lvert V\rvert + \ln \frac{1}{\delta}\right)\right)$	Hoeffding's ineq., Union bound
[Riondato and Kornaropoulos 2016] [Bergamini and Meyerhenke 2016]	shortest paths	$O\left(\frac{1}{\varepsilon^2}\left(\log_2 \mathsf{VD}(G) + \ln \frac{1}{\delta}\right)\right)$ †	VC-Dimension
This work	pairs of nodes	Variable, at most $O\left(\frac{1}{\varepsilon^2}\left(\log_2 \mathsf{L}(G) + \ln \frac{1}{\delta}\right)\right)$ ‡	Rademacher Avg., Pseudodimension

* See Def. 3.2 for the formal definition.
† $\mathsf{VD}(G)$ is the vertex diameter of the graph G.
‡ $\mathsf{L}(G)$ is the size of the largest weakly connected component of G. See Sect. 4.2 for tighter bounds.

Table of Contents

Experimental Evaluation

- Datasets
  - They use graphs of various nature (communication, citations, P2P, and social networks) from the SNAP repository
- The performance of the algorithm is measured using
  - runtime
  - sample size
  - accuracy
- Baselines compared
  - BA [Brandes 2001] - exact algorithm computing BC
  - RK [Riondato and Kornaropoulos 2016]

Results

Graph	ε	Runtime (sec.)	Speedup w.r.t. BA	Speedup w.r.t. RK	Runtime Breakdown (%) Sampling	Runtime Breakdown (%) Stop Cond.	Runtime Breakdown (%) Other	Sample Size	Reduction w.r.t. RK	Absolute Error ($\times 10^5$) max	Absolute Error ($\times 10^5$) avg	Absolute Error ($\times 10^5$) stddev
Soc-Epinions1	0.005	483.06	1.36	2.90	99.983	0.014	0.002	110,705	2.64	70.84	0.35	1.14
Directed	0.010	124.60	5.28	3.31	99.956	0.035	0.009	28,601	2.55	129.60	0.69	2.22
$\lvert V\rvert = 75,879$	0.015	57.16	11.50	4.04	99.927	0.054	0.018	13,114	2.47	198.90	0.97	3.17
$\lvert E\rvert = 508,837$	0.020	32.90	19.98	5.07	99.895	0.074	0.031	7,614	2.40	303.86	1.22	4.31
	0.025	21.88	30.05	6.27	99.862	0.092	0.046	5,034	2.32	223.63	1.41	5.24
	0.030	16.05	40.95	7.52	99.827	0.111	0.062	3,668	2.21	382.24	1.58	6.37
P2p-Gnutella31	0.005	100.06	1.78	4.27	99.949	0.041	0.010	81,507	4.07	38.43	0.58	1.60
Directed	0.010	26.05	6.85	4.13	99.861	0.103	0.036	21,315	3.90	65.76	1.15	3.13
$\lvert V\rvert = 62,586$	0.015	11.91	14.98	4.03	99.772	0.154	0.074	9,975	3.70	109.10	1.63	4.51
$\lvert E\rvert = 147,892$	0.020	7.11	25.09	3.87	99.688	0.191	0.121	5,840	3.55	130.33	2.15	6.12
	0.025	4.84	36.85	3.62	99.607	0.220	0.174	3,905	3.40	171.93	2.52	7.43
	0.030	3.41	52.38	3.66	99.495	0.262	0.243	2,810	3.28	236.36	2.86	8.70
Email-Enron	0.010	202.43	1.18	1.10	99.984	0.013	0.003	66,882	1.09	145.51	0.48	2.46
Undirected	0.015	91.36	2.63	1.09	99.970	0.024	0.006	30,236	1.07	253.06	0.71	3.62
$\lvert V\rvert = 36,682$	0.020	53.50	4.48	1.05	99.955	0.035	0.010	17,676	1.03	290.30	0.93	4.83
$\lvert E\rvert = 183,831$	0.025	31.99	7.50	1.11	99.932	0.052	0.016	10,589	1.10	548.22	1.21	6.48
	0.030	24.06	9.97	1.03	99.918	0.061	0.021	7,923	1.02	477.32	1.38	7.34
Cit-HepPh	0.010	215.98	2.36	2.21	99.966	0.030	0.004	32,469	2.25	129.08	1.72	3.40
Undirected	0.015	98.27	5.19	2.16	99.938	0.054	0.008	14,747	2.20	226.18	2.49	5.00
$\lvert V\rvert = 34,546$	0.020	58.38	8.74	2.05	99.914	0.073	0.013	8,760	2.08	246.14	3.17	6.39
$\lvert E\rvert = 421,578$	0.025	37.79	13.50	2.02	99.891	0.091	0.018	5,672	2.06	289.21	3.89	7.97
	0.030	27.13	18.80	1.95	99.869	0.108	0.023	4,076	1.99	359.45	4.45	9.53

Figure: Runtime, speedup, breakdown of runtime, sample size, reduction, and absolute error