MS&E 317/CS 263: Algorithms for Modern Data Models, Spring 2014
`http://msande317.stanford.edu`.
Instructors: Ashish Goel and Reza Zadeh, Stanford University.

Lecture 17, 6/2/2014. Scribed by Natasha Whitney.

18 Social Search: Quick Estimate for Distance Between Nodes

18.1 Problem Statement

Goal: Develop an algorithm to provide us with a quick estimate for distance between two nodes in a graph with an accuracy guarantee within 10%.

Sketch: We want to estimate the distance between nodes u, v. To do so, we will create a set, S, of landmarks nodes that we select uniformly and at random, and keep track of which nodes in S are closest to u and v (where $d(.,.)$ is defined as the shortest path distance in G in the specified metric space):

$$d(v, S) = \min_{w \in S} d(v, w)$$

Then, $d(v, w) \approx d(v, S) + d(w, S)$.

18.2 Algorithm for Distance Sketch

Definition: Given $G = (V, E)$ undirected, unweighted graph, define the algorithm as follows:

1. Sample sets S_i of size 2^i from the set of all nodes V uniformly at random without replacement, where i ranges from 1 to $\lfloor logN \rfloor$. Note, $|\Sigma S_i| = 2N$.

2. For each landmark set, S_i, find the closest landmark to v $l_i(v) = arg\min_{w \in S}\{d(v, w)\}$. Therefore, for all i we define our i-th distance estimate $\tilde{d_i}(v) = d(v, l_i(v))$. The sketch for v is defined as: $\{< l_0(v), \tilde{d}_{0(v)} >, ... < l_{\lfloor logN \rfloor}(v), \tilde{d}_{\lfloor logN \rfloor}(v) >\}$.

   Note, if a node is in S_i, its landmark is itself ($l_i(v) = v$) and $\tilde{d_i}(v) = 0$.

3. If u and v have the same landmark in set S_i then the distance between u and v is at most $D_i(u) + D_i(V)$ by the triangle inequality for metric spaces. Otherwise, S_i provides no information for pair (u, v). The estimate for the distance between u and v is then: $\tilde{d}(u, v) = \min_{i:l_i u = l_i v}\{\tilde{d_i}(u) = \tilde{d_i}(v)\}$

Note, steps (1) and (2) can be preprocessed. Step (3) is a query for a distance estimate between two nodes. Step (3) will always return a non-zero $\tilde{d}(v, w)$ because at very least $l_0(v) = l_0(u)$ ($|S_0| = 1$).

18.3 Accuracy and Efficiency

Here, we sketch the proof the following three claims about the accuracy and efficiency of the distance sketching algorithm. For a full proof, refer to actual paper [1]:

Claim 1: $\tilde{d}(u,v) \geq d(u,v)$
Proof of Claim 1: This holds by the triangle inequality for $d(.,.)$ defined in metric spaces.

Claim 2: $l_i(v)$ and $\tilde{d_i}(v)$ can be computed using $\lfloor logN \rfloor$ breadth-first-searches.
Proof of Claim 2: Given S_i, $\forall\, v \,\in\, V$ can compute $l_i(v)$ and $\tilde{d_i}(v)$ using one BFS as follows:

- $\forall\, w \,\in\, S_i$, $l_i(w) = w \;\&\; \tilde{d_i}(w) = 0$
- First step, keep track of all nodes, w_1 one hop away from $w \,\in\, S_i$. For all such nodes w_1, set landmark from S_i and $\tilde{d_i}(w_1) = 1$. Mark $\{w_1\}$ as visited, not explored.
- Second step, keep track of all nodes, w_2 one hop away from $\{w_1\}$. For all such nodes $w_2 \notin \{w_1\}$, set landmark as landmark of parent node in $\{w_1\}$ and $\tilde{d_i}(w_2) = 2$. Mark $\{w_2\}$ as visited, not explored.
- Repeat until all nodes $v \,\in\, V$ explored

And, as $i \,\in\, \{1, \lfloor logN \rfloor\} \rightarrow \lfloor logN \rfloor$ BFS required. Also, note that this BFS can be conducted as semi-streaming algorithm, stream through M edges, O(N) space required.

Claim 3: $\exists\, c > 0\, : Pr[\tilde{d}(u,v) \geq 2\,\lfloor logN \rfloor * d(u,v)] \leq c$
With such a c, to improve performance of algorithm repeat process. In particular, repeat algorithm $\frac{2\lfloor logN \rfloor}{c}$ times then $Pr[\tilde{d}(u,v) \geq 2\,\lfloor logN \rfloor * d(u,v)]$ for all times $\leq (1-c)^{\frac{2\lfloor logN \rfloor}{c}} \leq e^{-2\lfloor logN \rfloor} \leq \frac{1}{N^2}$

Intuition for Claim 3: For nodes that are close, landmark estimate comes from dense set (small neighborhood). For nodes that are far, landmark for distance estimates comes from sparse set (i.e. one random node will do okay).
Proof of Claim 3:

1. For $r = 1, 2, ..., \lfloor logN \rfloor$ and $d = d(u,v)$ define:

   $A_r = \{x : d(u,x) \leq rd\}$ all nodes in V at most rd away from u

   $B_r = \{x : d(v,x) \leq rd\}$ all nodes in V at most rd away from v

   Note $\forall i\, |A_r \cap B_r| \geq 2$ because $v \in A_i, B_i$ and $w \in A_i, B_i$

2. Case Analysis: Assume there exists an $r \in \{1, ..., \lfloor logN \rfloor\}$ such that $\frac{|A_r \cap B_r|}{|A_r \cup B_r|} \geq \frac{1}{2}$. If $\exists k$ such that $|S_k \cap (A_r \cap B_r)| = 1$ and $|S_k \cap (A_r \cup B_r)| = 1$ then $l_k(u) \in A_r \cap B_r$ and $l_k(v) \in A_r \cap B_r$ $\Rightarrow \tilde{d}(u,v) \leq 2r * d(u,v)$ and therefore $\tilde{d}(u,v) = arg\min \tilde{d}_k(u,v) \leq 2logN * d(u,v)$. What is the probability there exists such a k?

   Define $k = \lfloor log(|A_r \cap B_r|) \rfloor$ for r in (1).

Consider S obtained by picking every node in V independently and uniformly at random with probability p (same as choosing without replacement).

$P[|S \cap (A_r \cup B_r)| = |S \cap (A_r \cup B_r)| = 1] = |A_r \cap B_r| * p * (1-p)^{|A_r \cap B_r|-1} \geq \frac{kp}{2}(1-p)^{k-1}$

$\Rightarrow$ if we can choose $p = \frac{1}{k} \rightarrow P[|S \cap (A_r \cup B_r)| = |S \cap (A_r \cup B_r)| = 1] \geq \frac{1}{2e}$

But, because as we construct S_i for $i \in 1, ..., \lfloor logN \rfloor$ we are trying every value of p.

So, $\exists i : P[|S_i \cap (A_r \cup B_r)| = |S_i \cap (A_r \cup B_r)| = 1] \geq c$

S_i constructed geometrically $\rightarrow c = \frac{1}{4e}$.

3. **Prove** $\exists r \in \{1, ..., \lfloor logN \rfloor\}$ such that (2) holds by contradiction.

   Observe that $A_r \cup B_r \subset A_{r+1} \cap B_{r+1}$ by triangle inequality.

   $\Rightarrow \forall r \in \{1, ..., \lfloor logN \rfloor\} \frac{|A_r \cap B_r|}{|A_{r+1} \cap B_{r+1}|} < \frac{1}{2}$ by 2

   $\Rightarrow |A_2 \cap B_2| > 2 * |A_1 \cap B_1| \; ... \; |A_{\lfloor logN \rfloor} \cap B_{\lfloor logN \rfloor}| > 2 * |A_{\lfloor logN-1 \rfloor} \cap B_{\lfloor logN-1 \rfloor}|$

   $\Rightarrow |A_{\lfloor logN \rfloor} \cap B_{\lfloor logN \rfloor}| > 2^{\lfloor logN-1 \rfloor} * |A_1 \cap B_1|$ by telescoping

   $\Rightarrow |A_{\lfloor logN \rfloor} \cap B_{\lfloor logN \rfloor}| > 4^{\lfloor logN-1 \rfloor}$ by $|A_1 \cap B_1| \geq 2$

   $\Rightarrow |A_{\lfloor logN \rfloor} \cap B_{\lfloor logN \rfloor}| > N$ contradiction ■.

Therefore, repeating the distance sketching algorithm provides an estimate of the distance between any two points that is upper bounded by $2logN * d(u,v)$ with an exponentially low probability of failure.

References

[1] A.D. Sarma,S. Gollapudi,M. Najork,R. Panigraphy *A Sketch-Based Distance Oracle for Web-Scale Graphs.* 3rd ACM International Conference on Web Search and Data Mining (WSDM), Associationg for Computing Machinery, Inc, February 2010.