Finding densest subgraph in a bi-partite graph

Dhara Shah
Sushil Prasad
Danial Aghajarian

Follow this and additional works at: https://scholarworks.gsu.edu/computer_science_technicalreports

Recommended Citation
https://scholarworks.gsu.edu/computer_science_technicalreports/1
Finding densest subgraph in a bi-partite graph

Dhara Shah\(^1\), Danial Aghajarian\(^1\), and Sushil Prasad\(^1\)

Department of Computer Science, Georgia State University, Atlanta 30303, USA
{dshah8@, daghajarian@cs, sprasad}@gsu.edu

Abstract. Finding the densest subgraph in a bi-partite graph is a polynomial time problem. Also, each bi-partite graph has a densest connected subgraph. In this paper, we first prove that each bi-partite graph has a densest connected subgraph. This proof is different than that of an undirected graph, since our definition of the density is different. We then provide a max-flow min-cut algorithm for finding a densest subgraph of a bi-partite graph and prove its correctness of this binary search algorithm.

Keywords: densest subgraph · bi-partite · max-flow · densest connected

1 Densest subgraph of a bi-partite graph

We observe that there can be multiple densest bi-partite subgraphs of a bi-partite graph. We produce the following proof for this.

Theorem 1. Let \(G(S_a, S_b, E(S_a, S_b))\), \(G(S_{a2}, S_{b2}, E(S_{a2}, S_{b2}))\) be bi-partite subgraphs, with \(S_a \cap S_{a2} = \phi, S_b \cap S_{b2} = \phi, E(S_a, S_{b2}) = \phi, E(S_{a2}, S_b) = \phi\).

Let \(|S_a| = a_1, |S_{a2}| = a_2, |S_b| = b_1, |S_{b2}| = b_2, |E(S_a, S_b)| = e_1, |E(S_{a2}, S_{b2})| = e_2\).

Let the density of these graphs defined by
\[
\rho(G(S_a, S_b, E(S_a, S_b))) = \frac{e_1}{\sqrt{a_1b_1}}
\]
\[
\rho(G(S_{a2}, S_{b2}, E(S_{a2}, S_{b2}))) = \frac{e_2}{\sqrt{a_2b_2}}
\]
\[
\rho(G(S_a \cup S_{a2}, S_b \cup S_{b2}, E(S_a, S_b) \cup E(S_{a2}, S_{b2}))) = \frac{e_1 + e_2}{\sqrt{(a_1 + a_2)(b_1 + b_2)}}
\]

Prove that \(\frac{e_1 + e_2}{\sqrt{(a_1 + a_2)(b_1 + b_2)}} \leq \max\{\frac{e_1}{\sqrt{a_1b_1}}, \frac{e_2}{\sqrt{a_2b_2}}\}\).

Proof. Without loss of generality, let \(\max\{\frac{e_1}{\sqrt{a_1b_1}}, \frac{e_2}{\sqrt{a_2b_2}}\} = \frac{e_1}{\sqrt{a_1b_1}}\).

This implies,
\[
\frac{e_1}{\sqrt{a_1b_1}} \geq \frac{e_2}{\sqrt{a_2b_2}} \iff e_2 \leq \frac{e_1}{\sqrt{a_1b_1}} \sqrt{a_2b_2}
\]

(1)

Now, under this assumption,
\[
\frac{e_1 + e_2}{\sqrt{(a_1 + a_2)(b_1 + b_2)}} \leq \max\{\frac{e_1}{\sqrt{a_1b_1}}, \frac{e_2}{\sqrt{a_2b_2}}\}
\]
\[
\iff \frac{e_1 + e_2}{\sqrt{(a_1 + a_2)(b_1 + b_2)}} \leq \frac{e_1}{\sqrt{a_1b_1}}
\]

(2)
Also, LHS of equation (2) =
\[
\frac{e_1 + e_2}{\sqrt{(a_1 + a_2)(b_1 + b_2)}} \leq \frac{e_1 + e_1\sqrt{a_2b_2}}{\sqrt{(a_1 + a_2)(b_1 + b_2)}}
\]
Because (1)
\[
= \frac{e_1(\sqrt{a_1b_1} + \sqrt{a_2b_2})}{\sqrt{a_1b_1}(a_1 + a_2)(b_1 + b_2)}
\]
Hence, if we prove
\[
\frac{e_1(\sqrt{a_1b_1} + \sqrt{a_2b_2})}{\sqrt{a_1b_1}(a_1 + a_2)(b_1 + b_2)} \leq \frac{e_1}{\sqrt{a_1b_1}} = \text{RHS of equation (2)}
\]
we prove (2).

Here,
\[
\frac{e_1(\sqrt{a_1b_1} + \sqrt{a_2b_2})}{\sqrt{a_1b_1}(a_1 + a_2)(b_1 + b_2)} \leq \frac{e_1}{\sqrt{a_1b_1}}
\]
\[
\Leftrightarrow (\sqrt{a_1b_1} + \sqrt{a_2b_2}) \leq \sqrt{(a_1 + a_2)(b_1 + b_2)}
\]
\[
\Leftrightarrow (\sqrt{a_1b_1} + \sqrt{a_2b_2})^2 \leq (a_1 + a_2)(b_1 + b_2)
\]
\[
\Leftrightarrow 2\sqrt{a_1b_1a_2b_2} \leq a_1b_2 + a_2b_1
\]
\[
\Leftrightarrow \sqrt{(a_1b_2)(a_2b_1)} \leq \frac{a_1b_2 + a_2b_1}{2}
\]
This is true since arithmetic mean of two non-negative real numbers is always greater than or equal to their geometric mean.

Hence
\[
\frac{e_1 + e_2}{\sqrt{(a_1 + a_2)(b_1 + b_2)}} \leq \frac{e_1(\sqrt{a_1b_1} + \sqrt{a_2b_2})}{\sqrt{a_1b_1}(a_1 + a_2)(b_1 + b_2)}
\]
\[
\leq \frac{e_1}{\sqrt{a_1b_1}} = \max \{ \frac{e_1}{\sqrt{a_1b_1}}, \frac{e_2}{\sqrt{a_2b_2}} \}
\]

2 Maxflow Densest Subgraph (MDS)

MDS algorithm finds a densest bi-partite subgraph of a Triple Network in polynomial time. Inspired by [2] and [1], we use the max-flow min-cut strategy to obtain the densest bi-partite subgraph.

Definition 1. (Maximum density of a Triple Network) In a Triple Network \(G(V_a, V_b, E_a, E_b, E_c)\), maximum density is \(\rho^* = \max_{S_a \subseteq V_a, S_b \subseteq V_b} \frac{|E_c(S_a, S_b)|}{\sqrt{|S_a||S_b|}}\).

Let \(G_c[S_a, S_b]\) be a bi-partite subgraph of the Triple Network \(G\). Consider the number \(\lambda\) for which \(|E_c(S_a, S_b)| - \lambda \sqrt{|S_a||S_b|} = 0\). \(\lambda\), thus the density of this graph, depends on ratio \(r = \frac{|S_a|}{|S_b|}\) and \(|E_c(S_a, S_b)|\). Ratio \(r\) can take at most
Finding densest subgraph in a bi-partite graph

Let \( G(V_a, V_b, E_c) \) be a Triple Network with \( V_a \neq \phi, V_b \neq \phi \). Let \( G'(V', E') \) be the weighted directed flow network constructed from this network.

**Theorem 2.** MDS algorithm yields a densest subgraph of the Triple Network.

**Proof.** Let \( G(V_a, V_b, E_a, E_b, E_c) \) be a Triple Network with \( V_a \neq \phi, V_b \neq \phi \). Let \( G'(V', E') \) be the weighted directed flow network constructed from this network.
Algorithm 1 is $O$ iterations. In other words, this non-trivial s-t cut, if exists, is minimal. Hence the value of this cut is less than the value of trivial cut. Let $G$ be the minimum s-t cut of this flow network. From figure 1(a), if $S = \{s\}$ and $T = V_a \cup V_b \cup \{t\}$, then the value this trivial cut is $2m(|V_a| + |V_b|)$. However, if $S = \{s\} \cup S_a \cup S_b$ and $T = \{V_a \setminus S_a\} \cup \{V_b \setminus S_b\} \cup \{t\}$ then the value of a cut in this flow network is

$$2m|V_a| + 2m|V_b| - \sum_{v_a \in V_a \setminus S_a} 2m - \sum_{v_b \in V_b \setminus S_b} 2m + \sum_{v_a \in S_a} (2m + \frac{\lambda}{\sqrt{r}}) + \sum_{\{v_a, v_b\} \in E} 2(2m + \lambda \sqrt{r} - 2d(v_b))$$

$$= 2m(|V_a| + |V_b|) + \lambda \sqrt{r} |S_b| + \frac{\lambda}{\sqrt{r}} |S_a| - 2|E_c(S_a, S_b)|$$

$$= 2m(|V_a| + |V_b|) - 2(|E_c(S_a, S_b)| - \lambda \sqrt{|S_a||S_b|})(\text{substitute } r = \frac{|S_a|}{|S_b|})$$

This non-trivial s-t cut, if exists, is minimal. Hence the value of this cut is less than the value of trivial cut. In other words,

$$2m(|V_a| + |V_b|) \geq 2m(|V_a| + |V_b|) - 2(|E_c(S_a, S_b)| - \lambda \sqrt{|S_a||S_b|})$$

$$\geq 0.$$

So if, for given values of $\lambda$ and $r$, the flow network renders a non-trivial s-t cut $S, T$; then the subgraph $S \setminus \{s\} = G_c[V_a, V_b]$ has density $\lambda$ such that $|E_c(S_a, S_b)| - \lambda \sqrt{|S_a||S_b|} \geq 0$. Which implies that $\rho(S_a, S_b) \geq \lambda$. Hence, maximum density of $G$ has to be higher than the current guess of $\lambda$. However, if the flow network renders a trivial s-t cut, no subgraph of $G$ has density $\lambda$ with given $r$. Hence, maximum density of $G$ has to be lower than current guess of $\lambda$. By repeating this process as a binary search, eventually we will find the smallest $\lambda$ with $|E_c(S_a, S_b)| - \lambda \sqrt{|S_a||S_b|} = 0$ for the given $r$. By iterating on possible values of $r$, the maximum value of such $\lambda$ is found. This value is maximum density and the corresponding subgraph is a densest subgraph of $G$.

**Theorem 3.** MDS algorithm is a polynomial time algorithm.

**Proof.** The density difference of any two subgraphs of a bi-partite graph $G_c[V_a, V_b]$ is

$$\frac{m}{\sqrt{v_1v_2}} - \frac{m'}{\sqrt{v_1'v_2'}} \geq \frac{1}{|V_a| |V_b|} \quad \text{with } 0 \leq m, m' \leq |E_c|, 1 \leq v_1, v_1' \leq |V_a|, 1 \leq v_2, v_2' \leq |V_b|.$$  

This guarantees that the search for maximum density in the range $(0, \sqrt{|V_a||V_b|})$ can be performed with step size $\frac{1}{|V_a| |V_b|}$, halting in $O(|V_a|^{3/2} |V_b|^{3/2})$ iterations.

Within each iteration of this binary search, the minimum cut of the flow graph is calculated in $O(|V_a| + |V_b|)^2(2(|V_a| + |V_b|) + |E_c|)$). Hence, the complexity of algorithm 1 is

$$O(|V_a|^{1.5} |V_b|^{1.5}).$$

Adding the cost of BFS for finding connected components in $G_a$ and $G_b$, the upper-bound still remains unchanged.
Algorithm 1 Maxflow Densest Subgraph (MDS)

Input: Triple Network \(G(V_a, V_b, E_a, E_b, E_c)\), with \(V_a \neq \emptyset, V_b \neq \emptyset\)

Output: A densest bi-partite subgraph \(G_c[S_a, S_b]\) of \(G\)

1. possible\_ratios = \(\{\frac{1}{j} | i \in [1, \cdots |V_a|], j \in [1, \cdots |V_b|]\}\)
2. densest\_subgraph = \(\emptyset\), maximum\_density = \(\rho(V_a, V_b)\)
3. for ratio guess \(r \in\) possible\_ratios do
4.  low \(\leftarrow \rho(V_a, V_b)\), high \(\leftarrow \sqrt{|V_a||V_b|} \), \(g = G_c[V_a, V_b]\)
5.  while high - low \(\geq \frac{1}{|V_a|^2|V_b|^2}\) do
6.    mid = \(\frac{high + low}{2}\)
7.    construct a flow graph \(G'\) as described in \((f_1) - (f_6)\) and find the minimum s-t cut \(S, T\)
8.    \(g' = S \setminus \{\text{source node } s\}\)
9.    if \(g' \neq \emptyset\) then
10.       \(g \leftarrow g'\)
11.       \(low = \max\{mid, \rho(g)\}\)
12.    else high = mid
13.  if maximum\_density < low then
14.    maximum\_density = low
15.    densest\_subgraph = \(g\)

References