Mining Large Dense Subgraphs

Ajitesh Srivastava
Department of Computer Science
University of Southern California
ajiteshs@usc.edu

Charalampos Chelmis
Ming Hsieh Department of Electrical Engineering
University of Southern California
chelmis@usc.edu

Viktor K. Prasanna
Ming Hsieh Department of Electrical Engineering
University of Southern California
prasanna@usc.edu

ABSTRACT

Several applications including community detection in social networks and discovering correlated genes involve finding large subgraphs of high density. We propose the problem of finding the largest subgraph of a given density. The problem is a generalization of the Max-Clique problem which seeks the largest subgraph that has an edge density of 1. We define an objective function and prove that its optimization results in the largest graph of given density. We propose an algorithm that finds the subgraph by running multiple local search heuristics with random restarts. For massive graphs, where running the algorithm directly may be intractable, we use a sampling technique that reduces the graph to a smaller one which is likely to contain large dense subgraphs. We evaluate our algorithm on multiple real life and synthetic datasets. Our experiments show that our algorithm performs as well as the state-of-the-art for finding large subgraphs of high density, while providing density guarantees.

Keywords

Social Networks; Dense subgraphs; Discrete Otimization

1. INTRODUCTION

Finding the largest clique in a given graph (the Max-Clique problem) is an NP-hard problem. Since large cliques are not very common in real-world graphs, finding dense subgraphs is more meaningful. In fact, extracting dense subgraphs from large graphs is a key primitive in a variety of application domains. For a given graph G, we define the problem of finding largest subgraph with density at least ρ, where $\rho \in (0, 1]$. Here, we define density as the number of edges divided by the number of edges in a clique of the same size. The Max-Clique problem is a special case of our formulation when $\rho = 1$. We propose an objective function which when minimized, is guaranteed to find the largest subgraph with the given density. To optimize the objective function, we provide an algorithm based on local search with random restarts. We evaluate its performance on real and synthetic datasets. We show that when compared with the state-of-the-art [1], our algorithm returns denser and larger subgraphs while also having the additive advantage of providing density guarantees.

2. METHODOLOGY

Given a graph $G(V, E)$ and a real number ρ, we wish to find the largest subgraph with a density at least ρ. It is easy to see that in any graph with at least one edge, there always exists a subgraph with a density at least $\rho \,\forall \rho \in (0, 1]$. Selecting a pair of nodes that are connected by an edge produces a subgraph which satisfies this condition. Therefore, unless all nodes are isolated, a solution always exists. To find the largest subgraph of a given density, we define the following objective function.

$$E_\rho(S) = -|S| + \lambda(\rho - p_S)\delta_{\rho > p_S} \, . \tag{1}$$

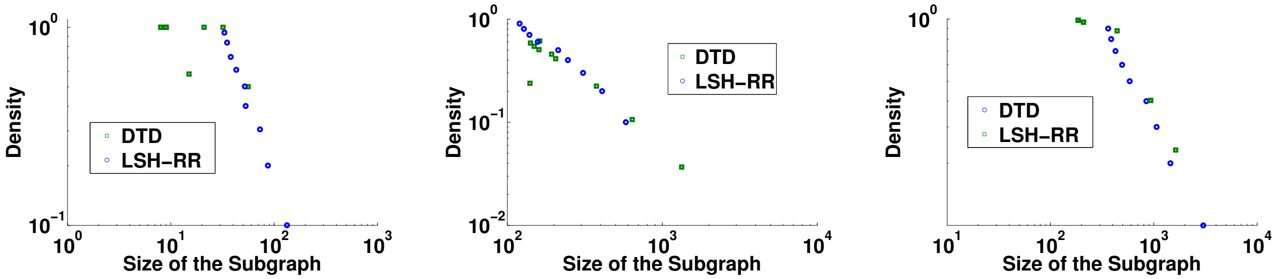
Here, $S \subseteq V$, p_S is the density of subgraph induced by S, and λ is a sufficiently large parameter.

Theorem 1. *For any $\epsilon > 0\,, \exists \lambda$ such that the subgraph induced by S that minimizes $E_\rho(S)$ is the largest subgraph in G with density $p_S > \rho - \epsilon$.*

We skip the proof for brevity. The following can be shown. If S^* is the subgraph with density p_{S^*} that minimizes E_ρ and $\lambda > |V|(|V|-1)(|V|-2)/4$, then 1) no bigger subgraph can have a higher density, and 2) either $p_{S*} > \rho$ or $\rho - p_{S*} < O(|V|^{-2})$.

To optimize E_ρ for a given graph, we use a local search heuristic with random restarts (Algorithm 1). We randomly initialize a population P of initial subsets of vertices. A local search begins from each of these subsets as follows: if adding a node improves (reduces) our objective function $E\rho$, then add that node to the current subset. Similarly, if removing a node improves E_ρ, then remove it from the subset. Continue adding and removing until no more nodes can be added or removed, or number iteration exceeds a pre-specified limit (I). This search is performed for all individuals in the population, resulting in N solutions, aggregated by selecting the subset with minimum $E\rho$.

WWW'16 Companion, April – 5, 20 , Montréal, Québec, Canada.
ACM 9 8 50 8/ /0 .
http://dx.doi.org/ 0. 5/28 25 8.2889559.



(a) HEPT (b) DBLP (c) Live Journal

Figure 1: Comparison of our method LSH-RR with state-of-the-art DTD for finding largest subgraph of given density.

Algorithm 1 Finding Dense Subgraphs using a Local Search Heuristic with Random Restarts

```
function LSH-RR($G$, $\rho$, $N$, $I$)
    $P[1 \dots N, 1 \dots |V|] \leftarrow Random\{0, 1\}$
    $j \leftarrow 0$
    for $n = 1 \rightarrow N$ do
        $m \leftarrow 1$
        $S \leftarrow toSet(P[n, :])$
        while $m < I$ or no change in $S$ do
            while $\exists$ $v$ adding which improves $E_\rho$ do
                $S \leftarrow S \cup v$
            end while
            while $\exists$ $v$ removing which improves $E_\rho$ do
                $S \leftarrow S \setminus v$
            end while
            $m \leftarrow m + 1$
        end while
        $P[n, :] = toVector(S)$
    end for
    $bestP \leftarrow$ Individual with the minimum $E_\rho$
    return $bestP$
end function
```

Table 1: Datasets used in our experiments

Dataset	# of nodes	# edges	# of nodes after reduction
HEPT	15,233	58,891	633
DBLP	425,957	1,049,866	3,066
Live Journal	4,036,538	34,681,189	4,103

3. EXPERIMETS

We conducted our experiments on three publicly available real-world datasets[1] summarized in Table 1.

3.1 Graph Reduction

To speed up computations on large graphs, we first reduce it into smaller one based on the assumption that all nodes in a large dense subgraphs have large degrees. If a node has a small degree, its removal is likely to increase the density of the subgraph. Therefore, given a large graph G, we remove its low degree nodes until all nodes in the remaining graph have degrees greater than a pre-defined δ.

3.2 Comparison

We run our Local Search Heuristic with Random Restarts (LSH-RR) on the reduced graph obtained from each dataset. The objective is to find the largest graph for a given density ρ. The comparison is against DTD [1] which takes a prameter α. Varying this α we were able to achieve different densities, however, no clear relationship was obtained between α and ρ. Figure 1 shows the comparison based on density of the subgraph obtained vs its size. A point on the top-right (high density, large subgraph) is desirable. We observe that LSH-RR either outperforms DTD or is very close to it. However, our method has the explicit parameter ρ that allows us to choose any desired density. We also conducted experiments on random Erdös-Renyi graphs of size 3000, where we planted a clique of size 30 and the task was to retrieve the planted clique. As reported in [1], DTD succeeded when edge probability in the graph was $p = 0.008$, but failed for $p = 0.1$ and $p = 0.5$. However, LSH-RR finds the clique for all the three cases.

4. CONCLUSION

We introduced the problem of finding the largest subgraph with density guarantees. We provided a theoretical insight into the proposed objective function, which we optimized based on an efficient local-search heuristic with random restarts. We evaluated our algorithm on real and synthetic large-scale graphs, showing that the subgraphs discovered by our method are larger and denser than subgraphs extracted by state-of-the-art. Our work leaves several open problems, such as the formal analysis of the graph reduction approach, and the design of more efficient randomized algorithms.

5. ACKNOWLEDGMENTS

This work was partially supported by U.S. National Science Foundation under grant NSF:5345039756. We would like to thank Dr. Rajgopal Kannan for his suggestions on the theorem.

6. REFERENCES

[1] C. Tsourakakis, F. Bonchi, A. Gionis, F. Gullo, and M. Tsiarli. Denser than the densest subgraph: extracting optimal quasi-cliques with quality guarantees. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 104–112. ACM, 2013.

[1] `https://snap.stanford.edu/data/index.html`