The Triangle-Densest-k-Subgraph Problem: Hardness, Lovász Extension, and Application to Document Summarization

Aritra Konar, and Nicholas D. Sidiropoulos

Department of Electrical and Computer Engineering, University of Virginia, VA, USA
(aritra,nikos)@virginia.edu

Abstract

We introduce the triangle-densest-k-subgraph problem (TDkS) for undirected graphs: given a size parameter k, compute a subset of k vertices that maximizes the number of induced triangles. The problem corresponds to the simplest generalization of the edge based densest-k-subgraph problem (DkS) to the case of higher-order network motifs. We prove that TDkS is NP-hard and is not amenable to efficient approximation, in the worst-case. By judiciously exploiting the structure of the problem, we propose a relaxation algorithm for the purpose of obtaining high-quality, sub-optimal solutions. Our approach utilizes the fact that the cost function of TDkS is submodular to construct a convex relaxation for the problem based on the Lovász extension for submodular functions. We demonstrate that our approaches attain state-of-the-art performance on real-world graphs and can offer substantially improved exploration of the optimal density-size curve compared to sophisticated approximation baselines for DkS. We use document summarization to showcase why TDkS is a useful generalization of DkS.

Introduction

The task of extracting dense subgraphs from a given graph has diverse applications in graph mining ranging from fraud detection (Hooi et al. 2016; Zhang et al. 2017), chemical informatics (Podolyan and Karypis 2009), computational biology (Saha et al. 2010) and knowledge discovery (Angel et al. 2012; Tixier, Malliaros, and Vazirgiannis 2016). Owing to its practical relevance, the problem has received extensive attention (see (Cadena, Chen, and Vullikanti 2018) and references therein) - we briefly highlight some prominent formulations.

Given an undirected graph, the classic densest-subgraph (DS) problem (Goldberg 1984) aims to detect the subgraph with the maximum average induced degree. The problem is known to be polynomial-time solvable, and admits a simple linear-time $1/2$ approximation via a greedy algorithm (Charikar 2000). However, real-world examples are known (Tsourakakis et al. 2013) where the greedy algorithm returns the trivial solution corresponding to the graph itself as the densest subgraph. This undesirable behavior can be attributed to the fact that the approach does not allow explicit specification of the desired subgraph size. Adding a simple size constraint to the DS problem results in the densest-k-subgraph (DkS) problem (Feige, Peleg, and Kortsarz 2001), which, for a specified node-size k, aims to find the subgraph with the maximum number of induced edges. Unfortunately, the constraint also renders DkS NP–hard. Moreover, the problem is notorious for being very difficult to approximate, in the worst-case sense (Khot 2006; Bhaskara et al. 2012; Manurangsi 2017). Notwithstanding such pessimistic results, polynomial-time algorithms which work well in practice for DkS are known – these include low (constant) rank matrix approximation techniques (Papailiopoulos et al. 2014) and a recent work (Konar and Sidiropoulos 2021), which uses tools from submodular optimization to construct a convex relaxation for DkS.

A salient feature of the aforementioned formulations is that they quantify subgraph density in terms of induced edges, which represent pair-wise relationships between vertices. However, real-world graphs are often rich in higher-order motifs, which signify stronger associations among vertices as compared to pair-wise relationships alone (Watts and Strogatz 1998). This suggests that leveraging higher-order motif structure for dense subgraph discovery can detect subgraphs which are more clique-like compared to those obtained via the edge-based formulations. For example, prior work (Tsourakakis 2015) has introduced the ℓ-clique densest subgraph problem to extract the subgraph with the largest average number of induced ℓ-cliques. This is a generalization of the DS problem (the latter corresponds to choosing $\ell = 2$) which remains polynomial-time solvable and also admits effective approximation via a greedy algorithm (Tsourakakis 2015). More importantly, applying this formulation with $\ell = 3$ (triangles, the simplest example of a higher-order motif) to real-world graphs yields subgraphs of higher *edge density* compared to using the DS with $\ell = 2$. However, like its edge-based counterpart, the ℓ-clique densest subgraph problem formulation does not provide a means of explicitly controlling the desired subgraph size. We argue that this is a restrictive feature, since it does not allow the end-user the flexibility in picking a desired solution. By varying the size explicitly, one can obtain small subsets of vertices which are tightly knit, to larger subsets which exhibit smaller density, and everything in-between.

For this purpose, in this paper, we introduce the triangle-densest-k-subgraph problem (TDkS). Given an undirected graph $\mathcal{G}$ on n vertices and a desired subgraph size k, we aim to compute the subgraph with the maximum number of induced triangles over all possible $\binom{n}{k}$ subgraphs. Clearly, TDkS is the simplest higher-order generalization of its edge based counterpart DkS. To the best of our knowledge, however, this is the first time that the problem has been studied. Can adopting such a formulation enable us to discover denser subgraphs compared to DkS, and thereby do a better job at exploring the optimal density-size curve on real-world graphs? Can TDkS extract more meaningful subsets than DkS in real-world applications? These are the main questions considered in our paper. Given this context, our contributions can be summarized as follows.

• **Hardness:** We prove that TDkS is NP-hard in the worst-case. Additionally, we show that it is difficult to obtain a favorable approximation of the optimal objective value of TDkS in polynomial-time.

• **Submodular relaxation and algorithm:** Not withstanding such pessimistic worst-case results, we focus on developing an approximation algorithm which can work well on real-world instances. We show and leverage the fact that the discrete cost function of TDkS is endowed with a specific type of combinatorial structure - namely, it is a subdmodular function. As such functions possess a unique, continuous, convex extension (i.e., the Lovász extension), we devise a convex relaxation for TDkS that minimizes the Lovász extension over the convex hull of the cardinality constraints. Additionally, a key technical contribution of our paper is to show that for TDkS, the Lovász extension admits an analytical functional form, which is difficult to determine for general submodular functions. We exploit this structure to develop a scalable Mirror Descent algorithm for solving the problem, which, combined with a simple rounding procedure, can be employed for extracting candidate triangle-dense subgraphs.

• **Experiments:** Our experiments reveal that the proposed approach is very effective in mining triangle-dense subgraphs on real-world datasets. Interestingly, it can also extract subgraphs of higher edge density than state-of-the-art DkS baselines, which is a bonus. Our experiments further indicate that when TDkS is used for unsupervised document summarization it yields more meaningful and interpretable summaries than DkS does. A sneak preview of our results can be found in Table 1, which depicts a summary of 30 words extracted from a text document representing a collection of reviews for the movie Joker, which we obtained from the review aggregator website https://metacritic.com. In order to apply our method to this text dataset, we constructed a graph-based representation where unique words in the document are vertices, with triplets of vertices being connected via triangles if the words they correspond to co-occur together in a tri-gram (i.e., a sub-sentence of length 3). Note that the TDkS represents a document summary composed of a subset of k words which co-occur most frequently in triples. An alternative graph model of text can be constructed from bi-grams - here, an edge connects a pair of vertices if the words they represent appear together in a bi-gram. In this context, the DkS is the subset of k words which appear most frequently in pairs. Since TDkS exploits higher-order co-occurrences (modeled using triangle motifs) relative to DkS, we intuitively expect that for a fixed summary size k, the former approach will yield more cohesive summaries compared to the latter. The results in Table 1 confirm this intuition.

Approach	Extracted Summary
TDkS	movie, joker, comic, book, character, reimagined, movies, one, anchored, performance, joaquin, phoenix, oscar, worthy, allows, would, study, depressing, engrossing, masterful, iconic, well, made, great, fine, almost, created, downside, killer, work.
DkS	film, movie, joker, comic, book, character, dc, much, work, performance, joaquin, phoenix, arthur, villain, todd, phillips, study, one, bad, social, last, well, point, also, go, enough, anything, specific, like, movies.

Table 1: Thirty-word summaries extracted from text reviews of the 2019 movie Joker by applying TDkS (resp. DkS) on a tri-gram (resp. bi-gram) graph-of-words model (Mihalcea and Tarau 2004).

We point out that, at a high level, our use of the Lovász relaxation for TDkS is in the spirit of (Konar and Sidiropoulos 2021) which introduced the Lovász relaxation for DkS. That being said, there are also important differences (apart from the fact that the two problems are distinct), the one key being that computing an analytical functional form for the Lovász extension of TDkS is substantially more challenging compared to the classical edge based case. Additionally, the form that the Lovász extension of TDkS takes is more complicated than that for DkS, which necessitates an entirely different algorithmic approach. Finally, to put our contributions into broader context, several recent works (Tsourakakis 2015; Tsourakakis, Pachocki, and Mitzenmacher 2017; Benson, Gleich, and Leskovec 2016; Zhang and Parthasarathy 2012) have considered generalizing classical edge-based graph mining tasks to account for higher-order network motifs. Our present work seeks to contribute to this thread of research by developing new tools for tackling a challenging problem in this area.

Primer on Submodularity

We provide a brief overview of basic concepts regarding submodular functions (Lovász 1983; Bach et al. 2013; Fujishige 2005). For a set of n objects $\mathcal{V} = \{1, \cdots, n\}$, a *set function* $F : 2^{|\mathcal{V}|} \rightarrow \mathbb{R}$ assigns a real value to any subset $\mathcal{S} \subseteq \mathcal{V}$. A set function F is said to be *submodular* if and only if $F(\mathcal{A} \cup \mathcal{B}) + F(\mathcal{A} \cap \mathcal{B}) \leq F(\mathcal{A}) + F(\mathcal{B})$ for all subsets $\mathcal{A}, \mathcal{B} \subseteq \mathcal{V}$. For the special case where $n = 2$ and $\mathcal{V} = \{a, b\}$, the above condition simplifies to $F(\emptyset) + F(\mathcal{V}) \leq F(\{a\}) + F(\{b\})$. A notable feature of submodular functions is that they possesses a continuous, *convex* extension known as the Lovász extension, which extends their domain from $2^{|\mathcal{V}|}$ to the unit interval $[0, 1]^n$ (recall

$n = |\mathcal{V}|$). Formally, the Lovász extension $f_L : [0,1]^n \rightarrow \mathbb{R}$ of a submodular function F is given by

$$f_L(\mathbf{x}) := \max_{\mathbf{g} \in \mathcal{B}_F} \mathbf{g}^T \mathbf{x}, \tag{1}$$

where the set $\mathcal{B}_F$ is the *base polytope* associated with F and is defined as

$$\mathcal{B}_F := \{\mathbf{g} \in \mathbb{R}^n : \mathbf{g}^T \mathbf{1}_{\mathcal{V}} = F(\mathcal{V});\ \mathbf{g}^T \mathbf{1}_{\mathcal{S}} \leq F(\mathcal{S}), \forall\, \mathcal{S} \subseteq \mathcal{V}\}. \tag{2}$$

It can be seen that the Lovász extension is the support function of the base polytope $\mathcal{B}_F$, and is thus a convex function. In fact, f_L is convex if and only if F is submodular. Furthermore, when evaluated at a binary vector $\mathbf{x} \in \{0,1\}^n$, the Lovász extension equals the value of the submodular function F.

Problem Statement

Given an undirected graph $\mathcal{G} := (\mathcal{V}, \mathcal{E})$ on n vertices and a parameter $4 \leq k < n$, we consider the problem of determining the subgraph of size k that exhibits the maximum weighted sum of induced triangles. Let $\mathcal{X} := \{\mathbf{x} \in \{0,1\}^n : \mathbf{1}^T \mathbf{x} = k\}$ be the set of all binary vectors with k non-zero entries. Formally, the triangle-densest-k-subgraph (TDkS) problem can be expressed as

$$\max_{\mathbf{x} \in \mathcal{X}} \left\{ f(\mathbf{x}) := \sum_{(u,v,w) \in \Delta} w_t x_u x_v x_w \right\}, \tag{3}$$

where Δ denotes the set of triangles in the graph (each counted once), and w_t is a positive weight associated with triangle $t := \{u, v, w\} \in \Delta$ [1]. We denote the optimal value of (3) as f^* (which represents the optimal weighted sum of induced triangles) and the optimal triangle density as $\rho_3^*(\mathcal{G}, k) := f^*/\binom{k}{3}$. Evidently, the above problem corresponds to maximizing a discrete third-order polynomial, subject to a cardinality constraint. This suggests that it is no easier to solve compared to the discrete quadratic maximization form associated with its classic edge-based counterpart, which is known to be NP–hard (Feige, Peleg, and Kortsarz 2001), and also very difficult to approximate. We make these notions concrete by proving the following pair of negative results regarding TDkS.

Theorem 1. *TDkS is NP–hard.*

In light of the above result, it is unlikely that the problem admits an efficient solution in polynomial-time. Consequently, we focus on developing effective approximation algorithms for problem (3) that run in polynomial-time. However, we first show that TDkS is fundamentally not amenable to favorable approximation in the worst-case sense; in fact it is more difficult to approximate compared to DkS.

More precisely, given an instance of DkS, let $\rho_2^*(\mathcal{G}, k)$ denote the optimal edge-density, defined as the ratio of the number of induced edges in the optimal size-k subgraph and the maximum possible number of induced edges $\binom{k}{2}$. Regarding the hardness of approximation of DkS, the following result is known (Manurangsi 2017).

[1] If $\mathcal{G}$ is unweighted, each triangle $t \in \Delta$ has weight $w_t = 1$.

Lemma 1. *Assuming that the Exponential Time Hypothesis (ETH) is valid, there is no polynomial-time algorithm that can approximate the optimal value of DkS better than a multiplicative factor $\alpha(n) := n^{1/(\log \log n)^c}$, where $c > 0$ is a universal constant.*

Note that the quantity $\alpha(n) > 1$. Hence, given an arbitrary instance of DkS, there is no polynomial-time algorithm which can output a size-k subgraph whose edge-density is guaranteed to be no worse than a fraction $1/(\alpha(n))^{1-\epsilon}$ of the optimal edge-density $\rho_2^*(\mathcal{G}, k)$, for any $\epsilon > 0$. In other words - if $\bar{\rho}_2(\mathcal{G}, k)$ denotes the edge density achieved by *any* polynomial-time approximation algorithm applied on a fixed instance of DkS, it must hold that

$$\rho_2^*(\mathcal{G}, k) \geq \bar{\rho}_2(\mathcal{G}, k) \geq O\left(\frac{1}{\alpha(n)}\right) \rho_2^*(\mathcal{G}, k). \tag{4}$$

We now demonstrate that the above hardness result for DkS can be utilized to derive an analogous hardness of approximation result for TDkS as well.

Theorem 2. *Assuming ETH is true, there is no polynomial-time algorithm that can approximate the optimal value of TDkS better than a multiplicative factor $\beta(n) := (\alpha(n))^{3/2}$.*

The above result implies that TDkS is more difficult to approximate compared to DkS, which is already known to be a challenging problem. Roughly speaking, Theorem 2 asserts that even the best possible polynomial-time approximation algorithm for TDkS must exhibit an approximation gap that grows as a polynomial in the size of the problem input n, which is a very pessimistic result.

That being said, the results of Theorem 1 and 2 are based on viewing the problem from the perspective of the worst-case scenario, which may not always arise in practice. With this in mind, we propose a convex relaxation for TDkS with the aim of obtaining high-quality, sub-optimal solutions on real-world instances.

The Lovász Relaxation

In order to explain our approach, we first reformulate TDkS in combinatorial form as follows. Let $\mathcal{C} := \{\mathcal{S} \subset \mathcal{V} : |\mathcal{S}| = k\}$ denote the collection of subsets of vertices of size k. Note that there is a one-to-one correspondence between the elements of $\mathcal{X}$ and $\mathcal{C}$; every vector $\mathbf{x} \in \mathcal{X}$ is precisely the indicator function of a subset of vertices $\mathcal{S} \in \mathcal{C}$, i.e., given a vector $\mathbf{x} \in \mathcal{X}$ and a set $\mathcal{S} \in \mathcal{C}$, we have the equivalence

$$x_u = \begin{cases} 1 & \Leftrightarrow u \in \mathcal{S} \\ 0 & \Leftrightarrow u \notin \mathcal{S}. \end{cases} \tag{5}$$

This observation allows us to equivalently express problem (3) in minimization form as

$$\min_{\mathcal{S} \in \mathcal{C}} \left\{ F(\mathcal{S}) := \sum_{(u,v,w) \in \Delta} F_{uvw}(\mathcal{S}) \right\}, \tag{6}$$

where for each triangle $(u, v, w) \in \Delta$, we have defined the function

$$F_{uvw}(\mathcal{S}) := F(\mathcal{S} \cap \{u, v, w\}) = \begin{cases} -w_t, & \text{if } (u, v, w) \in \mathcal{S}, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

Hence, the cost function $F(\mathcal{S})$ linearly decomposes over the set of triangles of $\mathcal{G}$, with each component function $F_{uvw}(\mathcal{S})$ contributing to the overall cost if and only if all three vertices constituting a triangle are included in the subgraph induced by $\mathcal{S} \in \mathcal{C}$. Our starting point is the following observation regarding the cost function $F(\mathcal{S})$.

Theorem 3. *$F(S)$ is a submodular function.*

Note that Theorem 3 does not change the fact that problem (3) is difficult to solve in the worst-case. However, it does allow us to adopt the following relaxation strategy. Let $\mathcal{P} := \{\mathbf{x} \in [0,1]^n; \mathbf{1}^T\mathbf{x} = k\}$ denote the convex hull of the combinatorial sum-to-k constraints. The key idea underpinning our approach is the following. Since the cost function of (6) is submodular, we can replace it by its Lovász extension to obtain the following equivalent problem

$$\begin{array}{ll} \min & f_L(\mathbf{x}) \\ \text{s.to} & \mathbf{x} \in \{0,1\}^n \cap \mathcal{P}. \end{array} \tag{8}$$

Note that the equivalence stems from the fact that the Lovász extension equals the value of $F(.)$ at all binary $\{0,1\}^n$ vectors. Upon dropping the discrete constraints, we obtain the relaxed problem

$$\min_{\mathbf{x} \in \mathcal{P}} \; f_L(\mathbf{x}) \tag{9}$$

which corresponds to minimizing the Lovász extension of F over the convex hull of the combinatorial set $\mathcal{C}$. Our rationale for employing the Lovász extension as a convex surrogate of F stems from the fact that it corresponds to the convex closure of F on the domain $[0,1]^n$. In other words, in a certain sense, the Lovász extension is the tightest convex under-estimator of F.

It is evident that problem (9) is convex, and hence can be optimally solved in polynomial-time to obtain a lower bound on the optimal value of (6). However, from an algorithmic standpoint, a major issue in solving the above problem is that the Lovász extension of a submodular function does not admit an analytical functional form in general. This can be attributed to the fact that the base polytope $\mathcal{B}_F$ is characterized by (potentially) an exponential number of inequalities in the problem dimension n. In a seminal paper, Edmonds (Edmonds 1970) established that a greedy algorithm based on sorting and querying F on n specific subsets suffices to compute a subgradient of the Lovász extension at any point $\mathbf{x} \in [0,1]^n$ without requiring explicit specification of the base polytope $\mathcal{B}_F$. While this result can be utilized within a projected subgradient framework for solving (9), for our present problem, we elect not to do so. This is due to the fact that the greedy algorithm is generic, i.e., it is not tailored to exploit the form of the submodular cost function of (6), which, in addition to its incremental nature, can result in a heavy computational footprint on large graphs.

We now demonstrate that it is possible to circumvent the aforementioned challenges related to solving (9) efficiently, and the main reason is that the Lovász extension for TDkS *does* admit an analytical form. In order to formally establish the result, we exploit the fact that F is linearly decomposable over the triangle-set Δ, which in turn implies that its base polytope can be expressed as the Minkowski sum of the base polytopes of the constituent functions F_{uvw}(Schrijver 2003, Theorem 44.6), i.e., we have

$$\mathcal{B}_F = \sum_{(u,v,w)\in\Delta} \mathcal{B}_{F_{uvw}}, \tag{10}$$

where $\mathcal{B}_{F_{uvw}}$ is the base polytope associated with the component F_{uvw}, and we have overloaded notation to represent set addition using the standard addition operator. Our next result shows that each such "sub"-polytope $\mathcal{B}_{F_{uvw}}$ admits a simple characterization.

Lemma 2. *The base polytope of F_{uvw} is given by*

$$\mathcal{B}_{F_{uvw}} = -w_t conv(\mathbf{e}_u, \mathbf{e}_v, \mathbf{e}_w). \tag{11}$$

Hence, the base polytope of F_{uvw} is the probability simplex in the space spanned by the coordinates indexed via (u,v,w) reflected about the origin and scaled by the weight w_t. Next, we exploit this result to derive an analytical form for the Lovász extension of F.

Theorem 4. *The Lovász extension of F is given by*

$$f_L(\mathbf{x}) = -\sum_{(u,v,w)\in\Delta} w_t \min\{x_u, x_v, x_w\}$$

The above result allows us to express problem (9) as

$$\min_{\substack{\mathbf{x}\in[0,1]^n,\\ \sum_{u=1}^n x_u = k}} \sum_{(u,v,w)\in\Delta} w_t \max\{-x_u, -x_v, -x_w\}, \tag{12}$$

which we designate as the Lovász relaxation. On inspecting the problem, however, it offers little in terms of an intuitive explanation as to why it can serve as a useful approximation for TDkS. To this end, our next result shows that the Lovász extension can be cast in an alternate form, which provides additional insight regarding (12).

Theorem 5. *The Lovász extension of F can be expressed as*

$$f_L(\mathbf{x}) = -\mathbf{t}^T\mathbf{x} + \sum_{(u,v,w)\in\Delta} w_t \cdot \phi_t(x_u, x_v, x_w) \tag{13}$$

where $\mathbf{t} \in \mathbb{R}^n$ denotes the vector of triangle degrees and $\phi_t(x_u,x_v,x_w) := \max\{x_u+x_v-2x_w, x_v+x_w-2x_u, x_u+x_w-2x_v\}$.

Using the above derived form, we now provide an intuitive explanation for the Lovász relaxation. Given any subset of vertices $\mathcal{S} \subseteq \mathcal{V}$, define the triangle "volume" $\text{vol}_\Delta(\mathcal{S}) := \sum_{v\in\mathcal{S}} t_v$ of $\mathcal{S}$ to be the sum of the weighted triangle counts of the vertices that constitute $\mathcal{S}$. Using a double counting argument, the triangle volume of any subset $\mathcal{S} \subseteq \mathcal{V}$ can be equivalently expressed as

$$\text{vol}_\Delta(\mathcal{S}) = t_1(\mathcal{S}) + 2t_2(\mathcal{S}) + 3t_3(\mathcal{S}), \tag{14}$$

where $t_1(\mathcal{S}), t_2(\mathcal{S})$ and $t_3(\mathcal{S})$ denote the weighted sum of triangles with one, two and three endpoints in $\mathcal{S}$, respectively. The above identity can be re-written as

$$t_3(\mathcal{S}) = (\text{vol}_\Delta(\mathcal{S}) - [2t_2(\mathcal{S}) + t_1(\mathcal{S})])/3, \forall\, \mathcal{S} \subseteq \mathcal{V}. \tag{15}$$

Note that the term on the left hand side corresponds to the objective function of TDkS. Hence, among subgraphs of a given size, those containing a large number of induced triangles must exhibit a large triangle volume (the first term on the right hand side) while simultaneously having few triangles being cut as a result of crossing the boundary of $\mathcal{S}$ (measured by the sum of the two terms subtracted from the volume). To be precise, for any given subset, a severed triangle with two endpoints $\{u, v\} \in \mathcal{S}$ affects the triangle counts (t_u, t_v) of both respective vertices (and hence the -2 factor), whereas a cut triangle with a single endpoint $u \in \mathcal{S}$ affects the triangle count t_u of only that vertex (and hence the -1 factor). The above equation asserts that for subgraphs with high triangle density, these losses stemming from severed triangles should be small compared to the triangle volume.

In order to establish the link with the form of the Lovász extension established in Theorem 5, we re-write (15) as

$$-t_3(\mathcal{S}) = -(\mathrm{vol}_\Delta(\mathcal{S}) + [2t_2(\mathcal{S}) + t_1(\mathcal{S})])/3. \qquad (16)$$

Since we have already established that $-t_3(\mathcal{S})$ is a submodular function and $\mathrm{vol}_\Delta(\mathcal{S})$ is a modular (and thus submodular) function, the remainder on the right hand side must also be submodular, as submodularity is preserved under addition. Furthermore, as the Lovász extension of the sum of submodular functions equals the sum of the Lovász extensions of the component functions, inspecting the result of Theorem 5 reveals that it corresponds to the sum of the Lovász extensions of the terms on the right hand side of (16). Hence, the extension preserves the first term, corresponding to the triangle volume, whereas it uses a convex surrogate for the second term to approximate the losses in the volume stemming from severed triangles. In particular, when solving the Lovász relaxation, each vertex is assigned a soft score that indicates how likely it is to belong to the triangle-densest-k subgraph. The formulation then assigns the highest emphasis on those vertices which have large triangle counts, but also exhibit small variation in scores across triangles.

Algorithm: Mirror Descent

In this section, we describe our algorithm for efficiently solving the Lovász relaxation (12), which is a convex problem. Since the Lovász extension is non-differentiable, this suggests employing a Euclidean projected subgradient algorithm for solving (12). The algorithm starts from an initial feasible point $\mathbf{x}^0 \in \mathcal{P}$ and then proceeds in the following iterative fashion

$$\mathbf{x}^{r+1} = \arg\min_{\mathbf{x}\in\mathcal{P}} \left\{ (\mathbf{g}^r)^T\mathbf{x} + \frac{1}{\beta^r}\|\mathbf{x}-\mathbf{x}^r\|_2^2 \right\}, \forall\, r \in \mathbb{N} \quad (17)$$

where $\mathbf{g}^r \in \partial f_L(\mathbf{x}^r)$ denotes a subgradient of the Lovász extension $f_L(\mathbf{x})$ at the current iterate $\mathbf{x} = \mathbf{x}^r$ and $\beta^r > 0$ is the learning rate. A standard result in convex optimization states that *if* the subgradients of f_L are bounded in the Euclidean sense, i.e., there exists a constant $G > 0$ such that

$$\|\mathbf{g}\|_2 \leq G, \forall\, \mathbf{g} \in \partial f_L(\mathbf{x}), \forall\, \mathbf{x} \in \mathcal{P}, \qquad (18)$$

then using the learning rate schedule $\beta_r = O(1/(\sqrt{r}))$ is sufficient to guarantee convergence to the optimal cost of (12) at a sublinear-rate of $O(G/\sqrt{r})$ (Bubeck 2015, Theorem 3.2). From this result, one can hope that the iteration complexity of the Euclidean subgradient algorithm is independent of the problem dimension n, which is a desirable trait for scaling up to large problem instances. However, the above claim is true provided that the Lipschitz constant G of the Lovász extension is independent of n. Unfortunately, this is not the case for our problem, as it can be shown that G varies like $O(n)$. Consequently, the Euclidean subgradient method (17) applied to solve (12) attains a dimension-*dependent* convergence rate of $O(\sqrt{\frac{n}{r}})$, which has undesirable implications for large-scale instances.

Thus, the non-Euclidean geometry of the problem renders the standard subgradient method (which measures distances in the ℓ_2-sense) a poor fit. In order to correct for this "mismatch" in geometry, we propose to employ the Mirror Descent algorithm (MDA) (Beck and Teboulle 2003), which can be viewed as a generalization of the subgradient algorithm to non-Euclidean spaces. To be specific, MDA is an iterative first-order algorithm that starts from a point $\mathbf{x}^0 \in \mathcal{P}$ and performs the following updates

$$\mathbf{x}^{r+1} = \arg\min_{\mathbf{x}\in\mathcal{P}} \left\{ (\mathbf{g}^r)^T\mathbf{x} + \frac{1}{\beta^r} D(\mathbf{x}, \mathbf{x}^r) \right\}, \forall\, r \in \mathbb{N} \quad (19)$$

where $D(.,.)$ is an appropriate "proximity"-measuring function. For example, on choosing $D(\mathbf{x}, \mathbf{x}^r) = \|\mathbf{x} - \mathbf{x}^r\|_2^2$, we obtain the standard subgradient algorithm. This proximal term can be viewed as the Bregman divergence associated with the function $\|\mathbf{x}\|_2^2$, which is strongly convex w.r.t. the ℓ_2 norm.

For our problem, the subgradients of the Lovász extension have constant size in the ℓ_∞ sense (see Section H of supplement), which motivates measuring distances using the ℓ_1 norm (which is the dual norm of the ℓ_∞ norm). This observation also suggests the choice of $D(.,.)$ in MDA to be the un-normalized Kullback-Leibler (KL) divergence between the points $\mathbf{x}$ and $\mathbf{x}^r$, which is defined as $D_{\mathrm{KL}}(\mathbf{x}, \mathbf{x}^r) = \sum_{i=1}^n x_i \left(\log \frac{x_i}{x_i^r} - 1 \right) + x_i^r$. Such a choice is based on the fact that the KL divergence is the Bregman divergence associated with the negative entropy function, which is strongly convex w.r.t. ℓ_1 norm on the feasible set $\mathcal{P}$. On performing the MDA udpates (19) using KL divergence with the learning rate schedule $\beta_r = O(\sqrt{\log n/r})$, invoking a standard result in convex optimization (Bubeck 2015, Theorem 4.2) guarantees a convergence rate of $O(G_\infty\sqrt{\log n/r})$, where G_∞ denotes the Lipschitz constant of f_L w.r.t. the ℓ_1 norm. Since this quantity is a constant, we obtain a convergence rate that exhibits a significantly improved dependence on the problem dimension n compared to that of the standard subgradient algorithm. Hence, fixing the geometry mismatch by employing the ℓ_1 norm to measure distances in MDA pays substantial dividends in this case. Owing to space constraints, the details of deriving the MDA updates are omitted. A full description of MDA is provided in Algorithm 1.

Since the computed solution $\mathbf{x}_L$ is not guaranteed to be integral in general, we perform a simple post-processing rounding step in order to obtain a binary indicator vector corresponding to a candidate subgraph. This is accomplished by simply projecting $\mathbf{x}_L$ onto the discrete sum-to-k constraints, Algorithm 1: MIRROR DESCENT

Input: Triangle list Δ, triangle weights $\{w_t\}_{t\in\Delta}$, subgraph size k, bisection tolerance $\epsilon > 0$.
Initialize: $\mathbf{x}^1 = (k/n)\mathbf{1}, r = 1$.

while Convergence criterion is not met **do**
  Obtain $\mathbf{g}^r \in \partial f_L(\mathbf{x}^r)$.
  Update step-size $\beta^r = c/\sqrt{r}$.
  $\mathbf{y}^r := \mathbf{x}^r \circledast \exp(-\beta^r \mathbf{g}^r)$.
  $\mathbf{x}^{r+1} = \text{BISECTION}(\mathbf{y}^r, k, \epsilon)$.
  Update $r = r + 1$.
end while
return $\mathbf{x}_L = (1/r)\sum_{i=1}^{r} \mathbf{x}^i$

Graph	n	m	$\lvert\Delta\rvert$
PPI-HUMAN	21,557	342K	2.39M
FACEBOOK-B	63,731	817K	3.51M
CAIDA	192K	609K	455K
WEB-STANFORD	281K	2.31M	11.33M
WEB-GOOGLE	875K	5.10M	13.39M
WIKI-TOPCATS	1.8M	28.51M	52.11M

Table 2: Summary of graph statistics: the number of vertices (n), the number of edges (m), and the number of triangles ($|\Delta|$).

which is equivalent to identifying the support of the k-largest entries in $\mathbf{x}_L$, and can be performed in $O(n \log k)$ time using heaps.

Experiments

In this section, we test the effectiveness of our proposed method in exploring the triangle density-size trade-off across a collection of real-world graphs. Additionally, we consider an application of TDkS to document summarization. Our results indicate that contrary to the worst-case scenario, real-world instances of TDkS can be far from adversarial, with the Lovász relaxation being effective at identifying high-quality, sub-optimal solutions.

Baselines

To the best of our knowledge, we are unaware of any pre-existing algorithms for the TDkS problem. Hence, we employ two state-of-the-art baselines for the (edge) densest-k-subgraph DkS problem, and test their efficacy at discovering triangle-dense subgraphs. These methods are described in brief below - additional details can be found in Section K of the supplement.

Lovász Relaxation for DkS (Konar and Sidiropoulos 2021): Similar to the approach considered herein (at a high level), but applied to the edge-density based formulation. Utilizes a variant of the Alternating Direction Method of Multipliers (ADMM) (Condat 2013) to solve the relaxed problem.

Low-rank Binary Matrix Principal Component (Papailiopoulos et al. 2014): Employs a rank-1 decomposition of the graph adjacency matrix $\mathbf{A}$, followed by solving the DkS problem with the rank-1 approximation in place of $\mathbf{A}$. The resulting problem admits a simple solution in $O(n)$ time which also provides an instance-specific upper bound on the optimal edge density for a given subgraph size. While not attainable in general, this bound can serve as a useful performance benchmark.

Triangle-density upper bound: The edge-density upper bound for DkS obtained via the above approach can also be converted into an upper bound on the optimal triangle density for TDkS via the Kruskal-Katona theorem (Kruskal 1963; Katona 1972). However, such a bound is not attainable in general for every choice of k as it is more loose compared to the bound on the optimal edge-density. In spite of this, we observed that on real-world graphs the Lovász relaxation for TDkS can attain this upper bound, or capture a significant fraction of it.

Since the first two baselines do not aim to directly detect triangle dense subgraphs, for fair comparison, we also compare the efficacy of our proposed methods for TDkS at detecting edge-dense subgraphs against the above baselines.

Datasets

We used a collection of graph datasets (summarized in Table 2) from standard repositories (Leskovec and Krevl 2014; Kunegis 2013) to test the performance of all methods. For TDkS, we used the well-known NODEITERATOR++ algorithm (Suri and Vassilvitskii 2011, Algorithm 2) to obtain a list of triangles in the graph, which incurs a run-time complexity of $O(|\mathcal{E}|^{3/2})$.

Results and Discussion

The outcomes of our experiments on the considered datasets are depicted in Figure 1. Our main findings are:

- With regard to subgraph triangle density (left column in Figure 1), solving the Lovász relaxation for TDkS via Mirror Descent followed by rounding consistently yields the best results across all considered graphs. In fact, for small subgraph sizes ($\leq$ 100), it is the only method that attains, or comes close to attaining the upper bound on the optimal triangle density. Our results demonstrate that although TDkS is NP–hard and difficult to approximate in the worst-case, the Lovász relaxation can still prove to be an effective tool for detecting triangle-dense subgraphs in real-world graphs.
- Although the Mirror Descent algorithm aims to detect subgraphs with high triangle density, it turns in a commendable performance in terms of edge density as well (right column in Figure 1). In fact, for subgraph sizes $\leq$ 200, it outperforms the dedicated edge based formulations, often by a significant margin and comes closest to attaining the edge density upper bound. This can be viewed as a consequence of the Kruskal-Katona theorem which formalizes the following intuitive notion: if a subgraph has high triangle density, then it must possess high edge density as well. Looking at Figure 1 (right column) confirms this observation.
- For large subgraph sizes, the edge density obtained by Mirror Descent / TDkS is often second (although by a small margin) to that obtained by applying the Lovász relaxation for DkS. Empirically, we note that this occurs (i.e., the blue

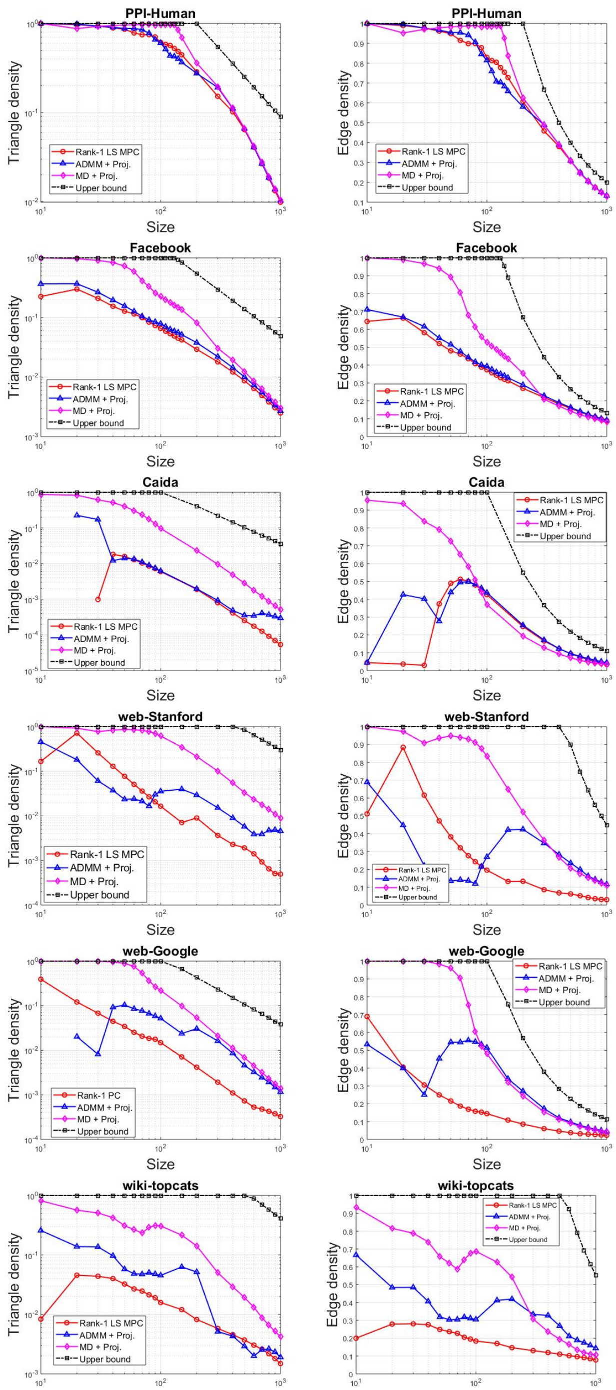

Figure 1: Left column: Triangle density (on a log-scale) vs size. Right column: Edge density vs size. Red (Rank-1 least-squares matrix principal component) and blue (ADMM) curves are methods for DkS while the magenta curve (MD) is for TDkS. The black curve in the right column depicts the edge density upper bound for DkS. This bound combined with the Kruskal-Katona theorem also yields an upper bound on TDkS (black curve in the left column).

curve "overtakes" the magenta curve) when the edge density falls below the 50% threshold. A possible explanation is as follows. Turán's theorem (Turán 1941) states that a graph can exhibit an edge density at most 0.5 without harboring any triangles. In other words, below this threshold, there do exist graphs with edge density up to 50% while containing very few triangles. Consequently, in the regime where the densest subgraph of a given size has edge density upper bounded by 0.5, employing a density measure based on edges may prove to be more beneficial as opposed to using triangles, if one cares more about edge density.

Conclusions

We considered the triangle-densest-k-subgraph problem (TDkS) which aims to compute the size k subgraph with the largest number of induced triangles. Unfortunately, not only is the problem NP–hard, but it is also difficult to approximate in polynomial-time, in the worst-case sense. With the aim of computing high-quality, sub-optimal solutions on real-world instances, we exploited the fact that the cost function of TDkS is submodular to construct a convex relaxation of the problem based on the Lovász extension of submodular functions. As we derived an analytical functional form for the extension, this enabled us to devise a Mirror Descent algorithm for efficiently solving the problem at scale. Our results on real-world graphs showcased that our approach can effectively exploit triangle motifs to attain state-of-the-art performance, and can provide a more effective means of exploring the density-size trade-off compared to baselines that only use edges for density maximization. Additionally, we utilized the problem of document summarization to showcase that TDkS can generate more informative word summaries compared to DkS.

Acknowledgements

Supported by the National Science Foundation and the Army Research Office under Grants No. IIS-1908070 and ARO-W911NF1910407 respectively. Aritra Konar acknowledges the assistance of Soumyajit Gupta in pre-processing the movie-review text datasets.

References

Angel, A.; Sarkas, N.; Koudas, N.; and Srivastava, D. 2012. Dense subgraph maintenance under streaming edge weight updates for real-time story identification. In *Proceedings of the 38th International Conference on Very Large Data Bases*, 574–585. VLDB Endowment.

Bach, F.; et al. 2013. Learning with Submodular Functions: A Convex Optimization Perspective. *Foundations and Trends® in Machine Learning*, 6(2-3): 145–373.

Beck, A.; and Teboulle, M. 2003. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3): 167–175.

Benson, A. R.; Gleich, D. F.; and Leskovec, J. 2016. Higher-order organization of complex networks. *Science*, 353(6295): 163–166.

Bhaskara, A.; Charikar, M.; Guruswami, V.; Vijayaraghavan, A.; and Zhou, Y. 2012. Polynomial integrality gaps for strong sdp relaxations of densest k-subgraph. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, 388–405. SIAM.

Bubeck, S. 2015. Convex Optimization: Algorithms and Complexity. *Foundations and Trends® in Machine Learning*, 8(3-4): 231–357.

Cadena, J.; Chen, F.; and Vullikanti, A. 2018. Graph anomaly detection based on Steiner connectivity and density. *Proceedings of the IEEE*, 106(5): 829–845.

Charikar, M. 2000. Greedy approximation algorithms for finding dense components in a graph. In *International Workshop on Approximation Algorithms for Combinatorial Optimization*, 84–95. Springer.

Condat, L. 2013. A primal–dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms. *Journal of Optimization Theory and Applications*, 158(2): 460–479.

Edmonds, J. 1970. Submodular functions, matroids, and certain polyhedra. *Edited by G. Goos, J. Hartmanis, and J. van Leeuwen*, 11.

Feige, U.; Peleg, D.; and Kortsarz, G. 2001. The dense k-subgraph problem. *Algorithmica*, 29(3): 410–421.

Fujishige, S. 2005. *Submodular functions and optimization*. Elsevier.

Goldberg, A. V. 1984. *Finding a maximum density subgraph*. Technical report, University of California Berkeley, CA.

Hooi, B.; Song, H. A.; Beutel, A.; Shah, N.; Shin, K.; and Faloutsos, C. 2016. Fraudar: Bounding graph fraud in the face of camouflage. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 895–904. ACM.

Katona, G. O. 1972. A simple proof of the Erdös-Chao Ko-Rado theorem. *Journal of Combinatorial Theory, Series B*, 13(2): 183–184.

Khot, S. 2006. Ruling out PTAS for graph min-bisection, dense k-subgraph, and bipartite clique. *SIAM Journal on Computing*, 36(4): 1025–1071.

Konar, A.; and Sidiropoulos, N. D. 2021. Exploring the Subgraph Density-Size Trade-off via the Lovaśz Extension. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, 743–751.

Kruskal, J. B. 1963. The number of simplices in a complex. *Mathematical optimization techniques*, 10: 251–278.

Kunegis, J. 2013. KONECT – The Koblenz Network Collection. In *Proc. Int. Conf. on World Wide Web Companion*, 1343–1350.

Leskovec, J.; and Krevl, A. 2014. SNAP Datasets: Stanford Large Network Dataset Collection. https://snap.stanford.edu/data. Accessed: 2021-08-15.

Lovász, L. 1983. Submodular functions and convexity. In *Mathematical programming the state of the art*, 235–257. Springer.

Manurangsi, P. 2017. Almost-polynomial ratio ETH-hardness of approximating densest k-subgraph. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, 954–961.

Mihalcea, R.; and Tarau, P. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, 404–411.

Papailiopoulos, D.; Mitliagkas, I.; Dimakis, A.; and Caramanis, C. 2014. Finding dense subgraphs via low-rank bilinear optimization. In *International Conference on Machine Learning*, 1890–1898.

Podolyan, Y.; and Karypis, G. 2009. Common pharmacophore identification using frequent clique detection algorithm. *Journal of chemical information and modeling*, 49(1): 13–21.

Saha, B.; Hoch, A.; Khuller, S.; Raschid, L.; and Zhang, X.-N. 2010. Dense subgraphs with restrictions and applications to gene annotation graphs. In *Annual International Conference on Research in Computational Molecular Biology*, 456–472. Springer.

Schrijver, A. 2003. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer Science & Business Media.

Suri, S.; and Vassilvitskii, S. 2011. Counting triangles and the curse of the last reducer. In *Proceedings of the 20th international conference on World wide web*, 607–614.

Tixier, A.; Malliaros, F.; and Vazirgiannis, M. 2016. A graph degeneracy-based approach to keyword extraction. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, 1860–1870.

Tsourakakis, C. 2015. The k-clique densest subgraph problem. In *Proceedings of the 24th International Conference on World Wide Web*, 1122–1132. International World Wide Web Conferences Steering Committee.

Tsourakakis, C.; Bonchi, F.; Gionis, A.; Gullo, F.; and Tsiarli, M. 2013. 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*, 104–112.

Tsourakakis, C. E.; Pachocki, J.; and Mitzenmacher, M. 2017. Scalable motif-aware graph clustering. In *Proceedings of the 26th International Conference on World Wide Web*, 1451–1460.

Turán, P. 1941. On an external problem in graph theory. *Mat. Fiz. Lapok*, 48: 436–452.

Watts, D. J.; and Strogatz, S. H. 1998. Collective dynamics of 'small-world' networks. *Nature*, 393(6684): 440.

Zhang, S.; Zhou, D.; Yildirim, M. Y.; Alcorn, S.; He, J.; Davulcu, H.; and Tong, H. 2017. Hidden: hierarchical dense subgraph detection with application to financial fraud detection. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, 570–578. SIAM.

Zhang, Y.; and Parthasarathy, S. 2012. Extracting analyzing and visualizing triangle k-core motifs within networks. In *2012 IEEE 28th international conference on data engineering*, 1049–1060. IEEE.