Efficient Centrality Maximization with Rademacher Averages

Leonardo Pellegrina

Dept. of Information Engineering, University of Padova

Padova, Italy

leonardo.pellegrina@unipd.it

ABSTRACT

The identification of the set of k most central nodes of a graph, or centrality maximization, is a key task in network analysis, with various applications ranging from finding communities in social and biological networks to understanding which seed nodes are important to diffuse information in a graph. As the exact computation of centrality measures does not scale to modern-sized networks, the most practical solution is to resort to rigorous, but efficiently computable, randomized approximations. In this work we present CentRA, the first algorithm based on progressive sampling to compute high-quality approximations of the set of k most central nodes. CentRA is based on a novel approach to efficiently estimate Monte Carlo Rademacher Averages, a powerful tool from statistical learning theory to compute sharp *data-dependent* approximation bounds. Then, we study the sample complexity of centrality maximization using the VC-dimension, a key concept from statistical learning theory. We show that the number of random samples required to compute high-quality approximations scales with finer characteristics of the graph, such as its vertex diameter, or of the centrality of interest, significantly improving looser bounds derived from standard techniques. We apply CentRA to analyze large real-world networks, showing that it significantly outperforms the state-of-the-art approximation algorithm in terms of number of samples, running times, and accuracy.

CCS CONCEPTS

• **Information systems → Data mining**; • **Mathematics of computing → Probabilistic algorithms**; • **Theory of computation → Sketching and sampling**.

KEYWORDS

Centrality Maximization, Rademacher Averages, Random Sampling

1 INTRODUCTION

Measuring the importance of nodes of a graph is a fundamental task in graph analytics [48]. To this aim, several *centrality* measures have been proposed to quantify the importance of nodes and *sets* of

nodes, such as the betweenness [28], closeness [6], pagerank [50], random walk [49], and harmonic [12] centralities. While the specific notion of centrality to use depends on the application [30], identifying important components of a network is crucial for many relevant tasks, such as finding communities in social and biological networks [31], identifying vulnerable nodes that may be attacked to disrupt the functionality of a network [37], and for influence maximization [38]. As in many applications the goal of the analysis is to identify important *sets* of central nodes, recent works proposed extensions and generalizations of centrality measures from single nodes to sets of nodes, such as the set betweenness [35], coverage [67], κ-path [1] and set closeness centrality [7]. A key problem is *set centrality maximization*, that consists in the task of finding the most central set of nodes of cardinality at most k, for some integer $k \geq 1$. For example, k may represents the budget of an attacker with the intention of disrupting a network, or the number of seed nodes to effectively diffuse relevant information in a graph.

To address the problems mentioned above, many algorithms have been proposed to compute centrality metrics exactly [18, 24]. For most centralities, however, these algorithms poorly scale when applied to large graphs [14, 57]. Therefore, often the only viable solution is to resort to approximate, but rigorous, estimates. In fact, several works (e.g., [14, 21, 54, 57, 58] for the betweenness centrality, see also Section 2) have recently proposed sampling approaches that provide approximations with rigorous guarantees of the centralities of individual nodes. In all these methods, the most critical challenge to address is to relate the *size* of the random sample with the *accuracy* of the approximation, i.e. studying the trade-off between the time to process the sample and the probabilistic guarantees that the sample provides w.r.t. the exact analysis. To achieve this goal, these works make use of sophisticated probabilistic and sampling techniques (described in more detail in Section 2).

However, as anticipated before, in many applications the interest of the analysis is on the centrality of *sets* of nodes, a task that is considerably more challenging, while yielding results that are substantially different. In fact, the centrality of a set of nodes is weakly related to the centralities of the individual nodes making part of the set. This implies that knowing the centralities of several nodes does not provide information on the centrality of the corresponding set of nodes as a whole. More importantly, all the most refined approaches mentioned above [14, 21, 54, 57, 58] are specifically tailored to provide guarantees for the centralities of individual nodes, and do not generalize to sets of nodes of size > 1. For this reason, they cannot be applied to centrality maximization or other tasks related to node sets.

Given the difficulty of set centralities approximation, this problem received only scant attention. In fact, only a few recent works proposed sampling-based algorithms to compute rigorous approximations of the centrality maximization task [43, 67]. As for the

individual nodes case, the main technical challenge of these methods is to relate the accuracy of the approximation to the *size* of the random sample to process, that impacts both the space and running time of the algorithm, imposing a severe trade-off. As the state-of-the-art techniques [43, 67] typically provide loose guarantees on the approximation quality, and require unrealistic a-priori knowledge of underlying graph, this problem remains difficult to address and computationally expensive in practice. Set centrality approximation is a challenging problem that requires new techniques to provide rigorous, yet efficiently computable, accuracy guarantees. This is the main goal of this work.

Our contributions. We introduce a new algorithm, called CentRA (Centrality Maximization with Rademacher Averages) for efficient centrality maximization using progressive sampling, that significantly outperforms previous works.

- CentRA is the first algorithm for approximate centrality maximization with data-dependent bounds and progressive sampling (Section 4.1 and 4.2). The progressive sampling strategy of CentRA makes it *oblivious* to the (unknown) set centrality of the optimal solution to approximate, in strong contrast with previous works (that instead requires a-priori knowledge of the optimal set centrality, see Section 3.1). To provide sharp data-dependent bounds, the central contribution at the core of CentRA is a new algorithmic strategy to estimate Monte Carlo Rademacher Averages, a key tool from statistical learning theory. CentRA enables the efficient computation of tight *graph-* and *data-dependent* probabilistic upper bounds to the Supremum Deviation, a key component of its strategy to provide rigorous high-quality approximations of set centralities.
- We study the sample complexity of centrality maximization with the VC-dimension, a fundamental concept from statistical learning theory. We derive new bounds to the number of sufficient samples to obtain high-quality approximations of set centralities with high probability (Section 4.3). Such bounds scale with granular properties of the graph or of the centrality measure of interest. Therefore, our novel bounds improve the analysis based on standard techniques (e.g., based on a union bound), typically offering a refined dependence on the graph size. In practice, these results can be naturally combined with the progressive sampling approach of CentRA by providing an upper limit to the number of samples it needs to process.
- We perform an extensive experimental evaluation, testing CentRA on several large real-world graphs (Section 5). Compared to the state-of-the-art, CentRA computes much sharper approximations for the same amount of work, or speeds up the analysis, of up to two orders of magnitude, for obtaining approximations of comparable quality.

2 RELATED WORK

We first introduce related work for centrality approximation of individual nodes. We focus on methods providing rigorous guarantees, a necessary requirement for many tasks and downstream analysis. Riondato and Kornaropoulos [57] study the VC-dimension [65], a key notion from statistical learning theory, of shortest paths to obtain approximation bounds for the betweenness centrality with random sampling. Riondato and Upfal [58] present an improved

method based on deterministic upper bounds to Rademacher Averages [40] and pseudodimension [55]. KADABRA [14] furtherly improved betweenness centrality approximations with adaptive sampling and a weighted union bound, while BAVARIAN [21] used Monte Carlo Rademacher Averages [5, 51] as a framework to fairly compare the accuracy of different estimators of the betweenness centrality. SILVAN [54] furtherly improved upon [21] leveraging non-uniform approximation bounds. As anticipated in Section 1, all these methods are specifically designed for individual nodes, and do not provide information or guarantees for node sets. Our algorithm CentRA leverages novel data-dependent bounds based on Monte Carlo Rademacher Averages. We remark that, while these techniques have been previously applied to centrality approximation of *individual* nodes [21, 54] and other pattern mining problems [51, 64], embedding and generalizing them to the context of *set* centrality approximation is extremely challenging, as we discuss in Section 3.2. The main technical contributions of CentRA are to prove new concentration results, that are general and may be of independent interest, and to develop a new algorithmic approach, based on the contraction principle of Rademacher Averages ([5, 41], see Section 4.1 of [16]) and progressive sampling to obtain efficiently computable, but still accurate, approximations for set centrality maximization. Interestingly, contraction approaches are fundamental theoretical tools that have been applied to prove generalization bounds for several complex Machine Learning tasks, such as multi-class classification and ranking [63], and models [5, 47], such as SVMs and neural networks [4, 20]. In this work we show their relevance and practical impact within graph analysis.

For the problem of approximate set centrality maximization, the state-of-the-art method is from Mahmoody et al. [43]. This work presents HEDGE, an algorithm based on a general framework for centralities that can be defined as submodular set functions, and that admit an appropriate randomized sampling oracle (an hyperedge sampler, see Section 3.1). In our work we adhere to a similar random sampling framework, but we develop a novel progressive and adaptive algorithm, and derive sharper sample complexity and data-dependent bounds. We show that our new algorithm CentRA significantly outperforms HEDGE [43] in terms of running time, accuracy, and approximation guarantees.

Other works considered efficient algorithms for different tasks or to approximate different notions of group centralities. [8] considers the problem of closeness centrality maximization. [45] studies the problem of maximizing the centrality of a set of nodes by adding new edges to the graph. [3] proposes a new group centrality measure inspired by Katz centrality that can be efficiently approximated. [2] studies the problem of approximating the group harmonic and group closeness centralities [11, 12]. [23] approximates percolation centrality with pseudodimension, while [19] is based on probability proportional to size sampling to estimate closeness centralities. Other recent works extended the computation of the betweenness centrality to dynamic [9, 10, 34], uncertain [60], and temporal networks [61].

3 PRELIMINARIES

In this Section we introduce the notation and the most important concepts for our algorithm CentRA.

3.1 Set Centralities

Let a graph $G = (V, E)$ with $n = |V|$ nodes. Let $C : 2^V \to \mathbb{R}$ be a *set centrality* function from the set of all possible subsets of V to $\mathbb{R}$. Let $\mathfrak{H} \subseteq 2^V$ be a space of sets of nodes, or *hyper-edges* [43, 67], such that each $h \in \mathfrak{H}$ is a subset of V. For a parameter $k \geq 1$, we define a family of functions $\mathcal{F}$ as $\mathcal{F} = \{f_{\mathcal{S}} : \mathcal{S} \subseteq V, |\mathcal{S}| \leq k\}$, where each $f_{\mathcal{S}} : \mathfrak{H} \to \{0, 1\}$ is a function from $\mathfrak{H}$ to $\{0, 1\}$ such that $f_{\mathcal{S}}(h) = 1$ if at least one of the nodes of $\mathcal{S}$ belongs to h, 0 otherwise; more formally, we have $f_{\mathcal{S}}(h) = \mathbb{1}[\mathcal{S} \cap h \neq \emptyset]$. Note that, to simplify notation, we will also denote f_u for all nodes $u \in V$ as equivalent of $f_{\{u\}}$. Under this setting, it is possible to define various set centralities $C(\cdot)$ as the average value of $f_{\mathcal{S}}$ over $\mathfrak{H}$

$$C(\mathcal{S}) = \frac{1}{|\mathfrak{H}|} \sum_{h \in \mathfrak{H}} f_{\mathcal{S}}(h),$$

using different notions of the space $\mathfrak{H}$. For instance, we define the *set betweenness centrality* by taking $\mathfrak{H}$ as the family of sets of nodes that are internal to shortest paths of the graph G, where each shortest path is from the node u to the node v, for all pairs of nodes $u, v \in V, u \neq v$. A large value of $C(\mathcal{S})$ denotes that many shortest paths of G pass along any of the nodes of $\mathcal{S}$. Analogously, for the κ-path centrality [1], $\mathfrak{H}$ is the family of sets of nodes traversed by simple paths of length at most κ; for the triangle centrality [43], $\mathfrak{H}$ contains sets of nodes incident to triangles of G. Intuitively, the centrality $C(\mathcal{S})$ of a node set $\mathcal{S}$ is large if the nodes of $\mathcal{S}$ "cover" a large fraction of the space $\mathfrak{H}$. Finally, note that $C(\mathcal{S})$ cannot be obtained from the values $\{C(\{u\}) : u \in \mathcal{S}\}$, i.e. the values of the centralities of individual nodes making part of the set $\mathcal{S}$.

The problem of *centrality maximization* is, for a given $k \geq 1$, to identify a set $\mathcal{S}^* \subseteq V$ of size at most k maximizing $C(\cdot)$, such that

$$\mathcal{S}^* = \underset{\mathcal{S} \subseteq V, |\mathcal{S}| \leq k}{\arg\max} \; C(\mathcal{S}).$$

As the computation of $\mathcal{S}^*$ is NP-Hard for most set centrality functions [26], the only viable solution is to obtain approximations that are efficient to compute, but with rigorous guarantees in terms of solution quality. We are interested in the following standard notion of α-approximation.

Definition 3.1. For any $\alpha > 0$, a set $\mathcal{S} \subseteq V$ with $|\mathcal{S}| \leq k$ provides an α-approximation of $\mathcal{S}^*$ if it holds $C(\mathcal{S}) \geq \alpha C(\mathcal{S}^*)$.

The state-of-the-art approach for the approximation of centrality maximization relies on the *submodularity* of the set centrality function [62]. The well known greedy algorithm that, starting from an empty set of nodes $\mathcal{S} = \emptyset$, iteratively picks and inserts in $\mathcal{S}$ a new node u^* maximizing the set centrality $u^* = \arg\max_u C(\mathcal{S} \cup \{u\})$, for k iterations, achieves an $(1 - 1/e)$-approximation. Such approximation ratio is essentially the best possible unless $P = NP$ [25, 26].

When considering large graphs, this approach is not efficient. In fact, simply computing the exact betweenness centrality of individual nodes requires $O(n|E|)$ time with Brandes' algorithm [18] (with a matching lower bound [13]); furthermore, each iteration of the greedy algorithm would require to evaluate and update the centralities $C(\mathcal{S} \cup \{u\})$ for nodes $u \notin \mathcal{S}$, which is clearly infeasible in reasonable time on large graphs. For this reason, approaches based on random sampling have been proposed [43, 67] to scale the approximation of set centralities.

Let a sample $\mathcal{H} = \{h_1, \ldots, h_m\}$ be a multiset of size m of hyper-edges from $\mathfrak{H}$, where each $h \in \mathcal{H}$ is taken independently and uniformly at random from $\mathfrak{H}$. The estimate $C_{\mathcal{H}}(\mathcal{S})$ of $C(\mathcal{S})$ for the set $\mathcal{S}$ computed on $\mathcal{H}$ is defined as

$$C_{\mathcal{H}}(\mathcal{S}) = \frac{1}{m} \sum_{s=1}^{m} f_{\mathcal{S}}(h_s).$$

The key observation of sampling approaches [43, 67] is that the greedy algorithm can be applied to a random sample, rather than considering the entire space $\mathfrak{H}$.

Let greedyCover$(k, \mathcal{H})$ be the algorithm defined as follows: starting from $\mathcal{S} = \emptyset$, at every iteration $i \in [1, k]$, insert the node $u^* = \arg\max_u C_{\mathcal{H}}(\mathcal{S} \cup \{u\})$ in $\mathcal{S}$; after all iterations, return $\mathcal{S}$.

It is clear that the *size m* of the random sample $\mathcal{H}$ imposes a trade-off between the *accuracy* of the approximation and the running time of greedyCover. Our goal is to precisely and sharply quantify this trade-off. First, note that, by definition, $C_{\mathcal{H}}(\mathcal{S})$ is an *unbiased* estimator of $C(\mathcal{S})$, as $\mathbb{E}_{\mathcal{H}}[C_{\mathcal{H}}(\mathcal{S})] = C(\mathcal{S}), \forall \mathcal{S} \subseteq V$. However, in order to provide guarantees in terms of solution quality, it is necessary to properly quantify $|C(\mathcal{S}) - C_{\mathcal{H}}(\mathcal{S})|$, i.e. the deviation of $C_{\mathcal{H}}(\mathcal{S})$ w.r.t. its expectation $C(\mathcal{S})$, for all $\mathcal{S} \subseteq V$.

A key quantity to study in order to control the approximation accuracy is the *Supremum Deviation (SD)* $\mathrm{D}(\mathcal{F}, \mathcal{H})$, defined as

$$\mathrm{D}(\mathcal{F}, \mathcal{H}) = \sup_{f_{\mathcal{S}} \in \mathcal{F}} |C(\mathcal{S}) - C_{\mathcal{H}}(\mathcal{S})|.$$

Note that it is not possible to evaluate $\mathrm{D}(\mathcal{F}, \mathcal{H})$ explicitly, as the values of $C(\mathcal{S})$ are unknown. Therefore, our goal is to obtain tight *upper bounds* to the SD $\mathrm{D}(\mathcal{F}, \mathcal{H})$.

The state-of-the-art method for approximate centrality maximization is HEDGE [43]. This algorithm computes, for an $\varepsilon \in (0, 1 - 1/e)$, an $(1 - 1/e - \varepsilon)$-approximation with high probability. More precisely, their analysis shows that if greedyCover$(k, \mathcal{H})$ is applied to a random sample $\mathcal{H}$ of size

$$m \in O\left(\frac{k \log(n) + \log(1/\delta)}{\varepsilon^2 C(\mathcal{S}^*)}\right), \quad (1)$$

then it holds $\mathrm{D}(\mathcal{F}, \mathcal{H}) \leq \varepsilon C(\mathcal{S}^*)/2$ with probability $\geq 1 - \delta$; moreover, this is a sufficient condition to guarantee that the output $\mathcal{S}$ of greedyCover$(k, \mathcal{H})$ is an $(1 - 1/e - \varepsilon)$-approximation of $\mathcal{S}^*$. The analysis of [43] (Lemma 2 and Theorem 1) combines the Chernoff bound (to upper bound the deviation $|C(\mathcal{S}) - C_{\mathcal{H}}(\mathcal{S})|$ of a set $\mathcal{S}$ with high probability) with an union bound over n^k events (an upper bound to the number of non-empty subsets of V of size $\leq k$).

As introduced in previous Sections, there are two key limitation of this approach. First, the union bound leads to loose guarantees, as it ignores any information of the input graph (apart from its size, which is very large for real-world graphs). Then, the number of samples to generate to compute an high-quality approximation of the most central set of nodes (given by (1)) depends on $C(\mathcal{S}^*)$, that is *unknown* a-priori.

Our new algorithm CENTRA tackles both these issues. Our first goal is to obtain a finer characterization of the trade-off between the size m of the random sample and bounds to the SD $\mathrm{D}(\mathcal{F}, \mathcal{H})$. CENTRA employs advanced and sharp *data-dependent* bounds to the SD based on Rademacher Averages (defined in Section 3.2), leading

to a much more efficient algorithm for approximate centrality maximization. Moreover, CENTRA is oblivious to the (unknown) values of $C(\mathcal{S}^*)$, computing an high-quality approximation *progressively* and *adaptively* with progressive sampling. Overall, CENTRA is the first progressive and data-dependent approximation algorithm for set centralities. While in this work we focus on the task of centrality maximization, CENTRA's bounds and rigorous guarantees may be useful to scale other exploratory analyses based on the centralities of sets of nodes. Furthermore, CENTRA directly applies to all set centrality functions supported by the framework introduced in Section 3.1.

3.2 Rademacher Averages

Rademacher averages are fundamental tools of statistical learning theory [5, 39, 40] and of the study of empirical processes [16]. We introduce the main notions and results relevant to our setting and defer additional details to [16, 46, 63]. While Rademacher averages are defined for arbitrary distributions and real valued functions, here we focus on the scenario of interest for set centralities.

The *Empirical Rademacher Average* (ERA) $\hat{\mathsf{R}}(\mathcal{F}, \mathcal{H})$ of the family of functions $\mathcal{F}$ computed on $\mathcal{H}$ is a key quantity to obtain a data-dependent upper bound to the supremum deviation $\mathsf{D}(\mathcal{F}, \mathcal{H})$. Let $\boldsymbol{\sigma} = \langle \sigma_1, \dots, \sigma_m \rangle$ be a vector of m i.i.d. Rademacher random variables, such that each entry σ_s of $\boldsymbol{\sigma}$ takes value in $\{-1, 1\}$ with equal probability. The ERA $\hat{\mathsf{R}}(\mathcal{F}, \mathcal{H})$ is defined as

$$\hat{\mathsf{R}}(\mathcal{F}, \mathcal{H}) = \mathbb{E}_{\boldsymbol{\sigma}} \left[\sup_{f_{\mathcal{S}} \in \mathcal{F}} \frac{1}{m} \sum_{s=1}^{m} \sigma_s f_{\mathcal{S}}(h_s) \right].$$

A central result in statistical learning theory implies that the *expected* supremum deviation is sharply controlled by twice the expected ERA, where the expectation is taken over the sample $\mathcal{H}$; this fundamental result is known as the *symmetrization lemma* (see Lemma 11.4 of [16] and Lemma A.3). However, the exact computation of $\hat{\mathsf{R}}(\mathcal{F}, \mathcal{H})$ is usually intractable, since it is not feasible to evaluate all the 2^m assignments of $\boldsymbol{\sigma}$. A natural solution to estimate the ERA is based on a Monte-Carlo approach [5].

For $t \geq 1$, let $\boldsymbol{\sigma} \in \{-1, 1\}^{t \times m}$ be a $t \times m$ matrix of i.i.d. Rademacher random variables. The *Monte-Carlo Empirical Rademacher average* (MCERA) $\mathsf{R}_m^t(\mathcal{F}, \mathcal{H}, \boldsymbol{\sigma})$ of $\mathcal{F}$ on $\mathcal{H}$ using $\boldsymbol{\sigma}$ is:

$$\mathsf{R}_m^t(\mathcal{F}, \mathcal{H}, \boldsymbol{\sigma}) = \frac{1}{t} \sum_{j=1}^{t} \sup_{f_{\mathcal{S}} \in \mathcal{F}} \left\{ \frac{1}{m} \sum_{s=1}^{m} \sigma_{js} f_{\mathcal{S}}(h_s) \right\}.$$

The MCERA is a powerful tool as it allows to directly estimate the expected supremum deviation from several random partitions of the data, also by taking into account the data-dependent structure of $\mathcal{F}$. Most importantly, it provides very accurate bounds even with a typically small number t of Monte Carlo trials (as we show in Section 5). For this reason, this Monte Carlo process leads to much more accurate bounds to the SD, compared to *deterministic upper bounds* (e.g., obtained with Massart's Lemma [44]) or other *distribution-free* notions of complexity, such as the VC-dimension.

While the MCERA has been applied to centrality approximation for individual nodes [21, 54], its generalization to centralities of

sets of nodes is highly non-trivial. In fact, for the case of set centralities, computing the MCERA is NP-Hard[1]. This is in strong contrast with the restriction to individual nodes ($k = 1$), where it can be computed efficiently. Moreover, the weighted sum $\sum_{s=1}^{m} \sigma_{js} f_{\mathcal{S}}(h_s)$ is neither monotone or submodular w.r.t. $\mathcal{S}$, due to the negative entries of $\boldsymbol{\sigma}$. For this reason, the greedy algorithm for constrained submodular function maximization described in Section 3.1 does not provide any approximation guarantee for the MCERA. It may be tempting to represent the function $\sum_{s=1}^{m} \sigma_{js} f_{\mathcal{S}}(h_s)$ as the difference $\sum_{s:\sigma_{js}=1} f_{\mathcal{S}}(h_s) - \sum_{s:\sigma_{js}=-1} f_{\mathcal{S}}(h_s)$ of two monotone submodular functions; unfortunately, there are strong inapproximability results for this maximization problem [36]. Moreover, we note that, while recent general approaches based on branch-and-bound optimization can be applied to compute the MCERA for general structured function families [51], they would still require a very expensive enumeration of the space of node subsets, incurring in high running times. These approaches cannot be adapted to centrality maximization as they would defy the purpose of using random sampling to scale the computation. These highly non-trivial computational barriers impose a significant challenge to obtaining accurate data-dependent approximations to set centralities.

In this work we tackle this problem with a new approach to upper bound the MCERA, that we introduce in Section 4.1. Our idea is to relate the ERA of arbitrary sets of nodes (of cardinality $\leq k$) to the ERA of linear combinations of *individual nodes*. We make use of the contraction principle [5, 16], a fundamental tool in statistical learning theory. This approach leads to accurate and efficiently computable probabilistic upper bounds to the ERA, key to our algorithm CENTRA.

4 CENTRA: EFFICIENT CENTRALITY MAXIMIZATION WITH RADEMACHER AVERAGES

This Section presents our contributions. In Section 4.1 we introduce new data-dependent bounds to the Supremum Deviation, the key techniques to our algorithm. In Section 4.2 we present our algorithm CENTRA for efficient centrality maximization with progressive sampling. In Section 4.3 we prove new sample complexity bounds for the rigorous approximation of centrality maximization.

4.1 Bounding the Supremum Deviation

In this Section we introduce our new, efficiently computable, data-dependent bounds to the SD, the key components of CENTRA.

For $t, m \geq 1$, recall that $\boldsymbol{\sigma} \in \{-1, 1\}^{t \times m}$ is defined as a $t \times m$ matrix of Rademacher random variables, such that, for all $s \in [1, m], j \in [1, t], \sigma_{js} \in \{-1, 1\}$ independently and with equal probability. We define the *Approximate Monte Carlo Empirical Rademacher Average* (AMCERA) $\tilde{\mathsf{R}}_m^t(\mathcal{F}, \mathcal{H}, \boldsymbol{\sigma})$ as

$$\tilde{\mathsf{R}}_m^t(\mathcal{F}, \mathcal{H}, \boldsymbol{\sigma}) = \frac{1}{t} \sum_{j=1}^{t} \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \leq k} \left\{ \frac{1}{m} \sum_{u \in \mathcal{S}} \sum_{s=1}^{m} \sigma_{js} f_u(h_s) \right\}.$$

Note that, as anticipated in Section 3.2, the AMCERA $\tilde{\mathsf{R}}_m^t(\mathcal{F}, \mathcal{H}, \boldsymbol{\sigma})$ takes the form of a linear combination of empirical deviations of

---

[1]Computing $\max_{\mathcal{S} \subseteq V, |\mathcal{S}| \leq k} C_{\mathcal{H}}(\mathcal{S})$ reduces to computing $\mathsf{R}_m^t(\mathcal{F}, \mathcal{H}, \boldsymbol{\sigma})$ by taking $t = 1$ and $\boldsymbol{\sigma} = \{1\}^{1 \times m}$.

individual nodes, rather than arbitrary sets of nodes. With Theorem 4.1 we show that the AMCERA $\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma)$ gives a sharp bound to the ERA $\hat{R}(\mathcal{F}, \mathcal{H})$, while Lemma 4.6 proves that we can compute it efficiently. A key quantity governing the accuracy of the probabilistic bound of Theorem 4.1 is the *approximate wimpy variance* $\tilde{w}_{\mathcal{F}}(\mathcal{H})$, that is defined as

$$\tilde{w}_{\mathcal{F}}(\mathcal{H}) = \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \leq k} \left\{ \frac{b_{\mathcal{H}}}{m} \sum_{u \in \mathcal{S}} \sum_{s=1}^{m} (f_u(h_s))^2 \right\},$$

where $b_{\mathcal{H}} = \max_{h \in \mathcal{H}} |h|$. We are now ready to state our new, efficiently computable, probabilistic bound to the ERA $\hat{R}(\mathcal{F}, \mathcal{H})$. Our proof (deferred to the Appendix for space constraints) is based on two key theoretical tools: a contraction inequality for Rademacher Averages [41] and a concentration bound for functions defined on the binary hypercube [16].

Theorem 4.1. *For $t, m \geq 1$, let $\sigma \in \{-1, 1\}^{t \times m}$ be a $t \times m$ matrix of Rademacher random variables, such that, for all $s \in [1, m], j \in [1, t]$, $\sigma_{sj} \in \{-1, 1\}$ independently and with equal probability. Then, with probability $\geq 1 - \delta$ over σ, it holds*

$$\hat{R}(\mathcal{F}, \mathcal{H}) \leq \tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma) + \sqrt{\frac{4\tilde{w}_{\mathcal{F}}(\mathcal{H}) \ln\left(\frac{1}{\delta}\right)}{tm}}.$$

We now investigate additional properties of the AMCERA. The following result proves guarantees on the expected approximation ratio of the AMCERA w.r.t. the MCERA. We note that, while there are no guarantees on the relation between the MCERA $R_m^t(\mathcal{F}, \mathcal{H}, \sigma)$ and the AMCERA $\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma)$ in the worst-case (i.e., that holds *for all* assignments of $\sigma \in \{-1, 1\}^{t \times m}$), their expectations (taken w.r.t. to σ) are tightly related. In fact, we prove that the *expected* AMCERA $\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma)$ is within a factor k of the ERA $\hat{R}(\mathcal{F}, \mathcal{H})$, therefore it provides a k-approximation of the MCERA $R_m^t(\mathcal{F}, \mathcal{H}, \sigma)$ in expectation. To make the dependence of the approximation bound on k explicit, define the set of functions $\mathcal{F}_j = \{f_{\mathcal{S}} : \mathcal{S} \subseteq V, |\mathcal{S}| \leq j\}$ (note that $\mathcal{F} = \mathcal{F}_k$).

Lemma 4.2. *It holds*

$$\hat{R}(\mathcal{F}_k, \mathcal{H}) \leq \mathbb{E}_{\sigma}\left[\tilde{R}_m^t(\mathcal{F}_k, \mathcal{H}, \sigma)\right] \leq k\hat{R}(\mathcal{F}_1, \mathcal{H}) \leq k\hat{R}(\mathcal{F}_k, \mathcal{H}).$$

In Section 4.2 we prove that the AMCERA can be computed efficiently, enabling the joint scalability and accuracy of CentRA.

We now prove that the greedy solution $\mathcal{S} = \text{greedyCover}(k, \mathcal{H})$ gives a sharp *upper bound* to the optimum set centrality $C(\mathcal{S}^*)$. Interestingly, the accuracy of this tail bound is *independent* of $\mathcal{F}$ and k, but leverages the concentration of the empirical estimator $\sup_{\mathcal{S}} C_{\mathcal{H}}(\mathcal{S})$ of the optimal set centrality $C(\mathcal{S}^*)$ directly. Our novel proof is based on its *self-bounding* property [16]. This result is instrumental for bounding the SD $D(\mathcal{F}, \mathcal{H})$ (Theorem 4.4) and for the stopping condition of CentRA, as we discuss in Section 4.2.

Theorem 4.3. *Let a sample $\mathcal{H}$ of size m and let $\mathcal{S}$ be the output of $\text{greedyCover}(k, \mathcal{H})$. With probability $\geq 1 - \delta$, it holds*

$$C(\mathcal{S}^*) \leq \frac{C_{\mathcal{H}}(\mathcal{S})}{1 - \frac{1}{e}} + \sqrt{\left(\frac{\ln\left(\frac{1}{\delta}\right)}{m}\right)^2 + \frac{2C_{\mathcal{H}}(\mathcal{S}) \ln\left(\frac{1}{\delta}\right)}{(1 - \frac{1}{e})m}} + \frac{\ln\left(\frac{1}{\delta}\right)}{m}.$$

The following result combines Theorem 4.1 with sharp concentration inequalities [16] to obtain a probabilistic bound to the

Supremum Deviation $D(\mathcal{F}, \mathcal{H})$ from a random sample $\mathcal{H}$ using the AMCERA. (proof deferred to the Appendix.)

Theorem 4.4. *Let $\mathcal{H}$ be a sample of m hyper-edges taken i.i.d. uniformly from $\mathfrak{H}$. For $k \geq 1$, let $\mathcal{S}$ be the output of $\text{greedyCover}(k, \mathcal{H})$. For any $\delta \in (0, 1)$, define $v, \tilde{R}, R,$ and η as*

$$v \doteq \frac{C_{\mathcal{H}}(\mathcal{S})}{(1 - \frac{1}{e})} + \sqrt{\left(\frac{\ln\left(\frac{5}{\delta}\right)}{m}\right)^2 + \frac{2C_{\mathcal{H}}(\mathcal{S}) \ln\left(\frac{5}{\delta}\right)}{(1 - \frac{1}{e})m}} + \frac{\ln\left(\frac{5}{\delta}\right)}{m},$$

$$\tilde{R} \doteq \tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma) + \sqrt{\frac{4\tilde{w}_{\mathcal{F}}(\mathcal{H}) \ln\left(\frac{5}{\delta}\right)}{tm}},$$

$$R \doteq \tilde{R} + \sqrt{\left(\frac{\ln\left(\frac{5}{\delta}\right)}{m}\right)^2 + \frac{2 \ln\left(\frac{5}{\delta}\right)\tilde{R}}{m}} + \frac{\ln\left(\frac{5}{\delta}\right)}{m},$$

$$\eta \doteq 2R + \sqrt{\frac{2 \ln\left(\frac{5}{\delta}\right)(v + 4R)}{m}} + \frac{\ln\left(\frac{5}{\delta}\right)}{3m}. \tag{2}$$

With probability at least $1 - \delta$ over the choice of $\mathcal{H}$ and σ, it holds $D(\mathcal{F}, \mathcal{H}) \leq \eta$.

In the following Section, we show that these results are essential to the theoretical guarantees of CentRA.

4.2 CentRA Algorithm

In this Section we present CentRA, our algorithm based on progressive sampling for centrality maximization. Algorithm 1 shows the pseudo-code of CentRA, that we now describe in more detail.

Algorithm 1: CentRA

Input: Graph $G = (V, E)$; hyper-edge space $\mathfrak{H}$; $k, t \geq 1$; $\varepsilon \in (0, 1 - \frac{1}{e})$; $\delta \in (0, 1)$.
Output: Set $\mathcal{S} \subseteq V$, s.t. $|\mathcal{S}| \leq k, C(\mathcal{S}) \geq (1 - \frac{1}{e} - \varepsilon)C(\mathcal{S}^*)$ with probability $\geq 1 - \delta$

1   $i \leftarrow 0; m_i \leftarrow 0; \mathcal{H}_i \leftarrow \emptyset$;
2   **while** *true* **do**
3     $i \leftarrow i + 1; \delta_i \leftarrow \delta/2^i$;
4     $m_i \leftarrow \text{samplingSchedule}()$;
5     $\mathcal{H}_i \leftarrow \mathcal{H}_{i-1} \cup \text{sampleHEs}(\mathfrak{H}, m_i - m_{i-1})$;
6     $\mathcal{S} \leftarrow \text{greedyCover}(k, \mathcal{H}_i)$;
7     $\sigma \leftarrow \text{sampleRadeVars}(t \times m_i)$;
8   **forall** $u \in V$ **do**
9   **forall** $j \in [1, t]$ **do** $r_j^u \leftarrow \frac{1}{m_i} \sum_{s=1}^{m_i} \sigma_{js} f_u(h_s)$;
10   **forall** $j \in [1, t]$ **do** $r_j \leftarrow \sup_{\mathcal{S}' \subseteq V, |\mathcal{S}'| \leq k} \left\{ \sum_{u \in \mathcal{S}'} r_j^u \right\}$;
11     $\tilde{R}_m^t(\mathcal{F}, \mathcal{H}_i, \sigma) \leftarrow \frac{1}{t} \sum_{j=1}^{t} r_j$;
12     $\xi \leftarrow \sqrt{\left(\frac{\ln\left(\frac{5}{\delta_i}\right)}{m_i}\right)^2 + \frac{2C_{\mathcal{H}_i}(\mathcal{S}) \ln\left(\frac{5}{\delta_i}\right)}{(1 - \frac{1}{e})m_i}} + \frac{\ln\left(\frac{5}{\delta_i}\right)}{m_i}$;
13     $\eta \leftarrow$ as in Equation (2) replacing δ by δ_i and $\mathcal{H}$ by $\mathcal{H}_i$;
14   **if** $(1 - \frac{1}{e}) \left[(1 - \frac{1}{e} - \varepsilon)\xi + \eta\right] \leq \varepsilon C_{\mathcal{H}_i}(\mathcal{S})$ **then return** $\mathcal{S}$;

CentRA takes in input a graph G, a space of hyper-edges $\mathfrak{H}$, and the parameters $k, t, \varepsilon,$ and δ, with the goal of computing an approximation $\mathcal{S}$ of $\mathcal{S}^*$ such that $C(\mathcal{S}) \geq (1 - 1/e - \varepsilon)C(\mathcal{S}^*)$ with probability $\geq 1 - \delta$.

As a progressive sampling-based algorithm, CentRA works in iterations. In each iteration i, CentRA considers a random sample $\mathcal{H}_i$ of size m_i, and tries to establish if it is possible to compute an high-quality approximation of $\mathcal{S}^*$ from $\mathcal{H}_i$. To verify whether this holds, CentRA checks a suitable stopping condition (defined below), returning the current solution when true.

In line 1, the algorithm initializes the variables i to 0, $\mathcal{H}_i$ to the empty set, and $m_i = |\mathcal{H}_i|$ to 0. Then, the iterations of the algorithm start at line 2. At the beginning of each iteration, i is increased by 1, and the confidence parameter δ_i is set to $\delta/2^i$. This correction is needed to prove the guarantees of the algorithm, i.e., to make sure that the probabilistic bounds computed at iteration i holds with probability $\geq 1 - \delta_i, \forall i$, so that CentRA is correct with probability $\geq 1 - \delta$ when stopping at any iteration i (more details in the proof of Proposition 4.5). Then, the number of samples m_i to consider is given by a procedure samplingSchedule (line 4). This procedure can be implemented in several ways. For example, the geometric progression defining $m_i = \alpha \cdot m_{i-1}$ for some $\alpha > 1$ is simple but considered to be optimal [56]. We combine this geometric progression (with $\alpha = 1.2$) with an adaptive schedule that tries to guess the next sample size using an optimistic target for the SD bound η; in fact, note that by setting R = 0 and $\nu = 1$ in Thm. 4.4, solving the equality (2) for m gives a lower bound to the minimum number m of samples to obtain a sufficiently small η.

After getting the number of samples m_i, the algorithm generates the new sample $\mathcal{H}_i$ composing $\mathcal{H}_{i-1}$ with new $m_i - m_{i-1}$ random hyper-edges, taken from $\mathfrak{H}$ by the procedure sampleHEs($\mathfrak{H}, m_i - m_{i-1}$). We describe the implementation of sampleHEs that we use for the set betweenness centrality in Section 5. Then, CentRA calls the procedure greedyCover($k, \mathcal{H}_i$), that finds a subset of nodes of size $\leq k$ covering many hyper-edges of $\mathcal{H}_i$ with the greedy procedure introduced in Section 3.1, returning the identified set $\mathcal{S}$.

In lines 7-11, CentRA computes the AMCERA (Section 4.1). To do so, it generates a matrix σ of size $t \times m_i$ by sampling tm_i Rademacher random variables (Section 3.2) using the procedure sampleRadeVars($t \times m_i$) (line 7). Then, for each vertex $u \in V$ and each trial of index $j \in [1, t]$, it populates the values of r_j^u as the weighted average of the function f_u on $\mathcal{H}_i$ w.r.t. the entries of the matrix σ (line 9). Using these values, CentRA computes r_j as the supremum sum of at most k entries of the set $\{r_j^u : u \in V\}$ (line 10).

CentRA obtains the AMCERA in line 11 as the average of the t values of r_j. We remark that it is not needed to generate the entire matrix σ at each iteration of the algorithm, and to compute r_j^u for all $u \in V$, as done in the simplified procedure described above; it is sufficient to extend σ with $m_i - m_{i-1}$ new columns, and to update the values of r_j^u *incrementally* as each sample is generated and added to $\mathcal{H}_i$. All these operations can be done efficiently, as proven by Lemma 4.6.

In line 12, the algorithm defines ξ, an accuracy parameter that quantifies how far is the estimate $C_{\mathcal{H}_i}(\mathcal{S})$ from the optimal centrality $C(\mathcal{S}^*)$: it holds $C(\mathcal{S}^*) \leq \frac{C_{\mathcal{H}_i}(\mathcal{S})}{1-1/e} + \xi$ with probability $\geq 1 - \delta_i/5$ (see Theorem 4.3 and the value of ν in the statement and proof of Theorem 4.4). After obtaining the AMCERA, CentRA computes (line 13) an upper bound η to the supremum deviation $D(\mathcal{F}, \mathcal{H}_i)$ on the sample $\mathcal{H}_i$ with confidence δ_i, using the result of Theorem 4.4.

Finally, CentRA checks a stopping condition (line 14); if true, it means that $\mathcal{S}$ provides an $(1 - 1/e - \varepsilon)$-approximation of the optimum $\mathcal{S}^*$ with enough confidence, therefore CentRA returns $\mathcal{S}$ in output. CentRA uses a new, improved stopping condition to establish its theoretical guarantees. As discussed previously, the condition $D(\mathcal{F}, \mathcal{H}) \leq \varepsilon C(\mathcal{S}^*)/2$ proposed by HEDGE [43] is sufficient to guarantee that $C(\mathcal{S}) \geq (1 - 1/e - \varepsilon)C(\mathcal{S}^*)$. However, this condition is not necessary. CentRA combines the upper bound $C(\mathcal{S}^*) \leq \nu = C_{\mathcal{H}_i}(\mathcal{S})(1 - 1/e)^{-1} + \xi$ to $C(\mathcal{S}^*)$ with the lower bound $C_{\mathcal{H}_i}(\mathcal{S}) - \eta \leq C(\mathcal{S})$ to $C(\mathcal{S})$, rather than using η twice (for both the upper bound to $C(\mathcal{S}^*)$ and the lower bound to $C(\mathcal{S})$, see the proof of Proposition 4.5). Note that, in general, we expect $\xi \ll \eta$, as ξ is *independent* of k and any property of $\mathcal{F}$ (e.g., its complexity), but sharply scales with the empirical estimate $C_{\mathcal{H}_i}(\mathcal{S})$ of $C(\mathcal{S}^*)$.

We now prove the correctness of CentRA, and a bound to its time complexity for *incrementally* processing the random samples and for computing the AMCERA. (the proofs are in the Appendix.)

PROPOSITION 4.5. *Let $\mathcal{S}$ be the output of CentRA. With probability* $\geq 1 - \delta$, *it holds* $C(\mathcal{S}) \geq (1 - \frac{1}{e} - \varepsilon)C(\mathcal{S}^*)$.

LEMMA 4.6. *Define $b_{\mathcal{H}}$ as $\sup_{h \in \mathcal{H}} |h| \leq b_{\mathcal{H}} \leq n$. When CentRA stops at iteration $i \geq 1$ after processing m_i samples, it computes $\tilde{R}_{m_j}^t(\mathcal{F}, \mathcal{H}_j, \sigma)$, for all $j \in [1, i]$, in time $O((b_{\mathcal{H}} m_i + k) t \log(n))$.*

Note that the time bound above considers the total number of operations performed by CentRA for computing $\tilde{R}_{m_j}^t(\mathcal{F}, \mathcal{H}_j, \sigma)$, summed over *all* iterations $j \leq i$. Notably, the time needed by CentRA to compute the AMCERA is linear in the final sample size m_i and logarithmic in the number of nodes n of the graph. Furthermore, the parameter $b_{\mathcal{H}}$ is typically very small, as we discuss in Section 4.3 (e.g., for the betweenness centrality is at most the *vertex diameter B* of the graph [57]). We obtain the total running time of CentRA by summing the bound from Lemma 4.6 and $O(T_{\mathfrak{H}} m_i + n \log(n))$, where $T_{\mathfrak{H}}$ is the time to sample one hyper-edge from $\mathfrak{H}$, and the second term is for the greedy algorithm greedyCover (using the implementation from [15], and assuming the final iteration index $i \in O(1)$, that is always the case in practice). In our experimental evaluation (Section 5) we show that CentRA is a very practical and scalable approach, since the time required to compute the AMCERA is negligible w.r.t. to the time to generate the random samples and greedyCover.

Finally, we remark that CentRA applies directly to approximate set centrality functions that are submodular and can be defined as averages over some hyper-edge space $\mathfrak{H}$ (see Section 3.1), such as κ-path, coverage, and triangle centralities, as CentRA is oblivious to the procedure sampleHEs to sample hyper-edges.

4.3 Sample Complexity bounds

In this Section we present a new bound (Theorem 4.7) to the sufficient number of random samples to obtain an $(1 - \frac{1}{e} - \varepsilon)$-approximation of $\mathcal{S}^*$. (due to space constraints, we defer all proofs of this Section to the Appendix.)

We first define the range space associated to set centralities and its VC-dimension, and remand to [46, 47, 63] for a more complete introduction to the topic. Define the range space $Q_k = (\mathfrak{H}, R_k)$, where $\mathfrak{H}$ is a family of hyper-edges, and

$$R_k = \{\{h : h \in \mathfrak{H}, f_{\mathcal{S}}(h) = 1\} : \mathcal{S} \subseteq V, |\mathcal{S}| \leq k\}$$

is a family of subsets of $\mathfrak{H}$, indexed by the indicator functions $f_S \in \mathcal{F}$ associated to subsets of V with size at most k. For any $\mathcal{H} \subseteq \mathfrak{H}$ of size m, the projection $P(\mathcal{H}, R_k)$ of R_k on $\mathcal{H}$ is

$$P(\mathcal{H}, R_k) = \{\mathcal{H} \cap r : r \in R_k\}.$$

We say that $\mathcal{H}$ is *shattered* by R_k if $|P(\mathcal{H}, R_k)| = 2^m$. The *VC-dimension* [65] $VC(Q_k)$ of the range space Q_k is the maximum cardinality m of a set $\mathcal{H}$ that is shattered by R_k. Under this framework, we are ready to state the following sample complexity bound.

THEOREM 4.7. *Let $\mathcal{H}$ be a sample of m hyper-edges taken i.i.d. uniformly from $\mathfrak{H}$. For $k \geq 1$, let S be the output of* greedyCover$(k, \mathcal{H})$. *Let d_k such that $VC(Q_k) \leq d_k$. For $\delta \in (0, 1)$ and $\varepsilon \in (0, 1 - \frac{1}{e})$, if*

$$m \in O\left(\frac{d_k \log\left(\frac{1}{C(S^*)}\right) + \log\left(\frac{1}{\delta}\right)}{\varepsilon^2 C(S^*)}\right), \tag{3}$$

then, it holds $C(S) \geq (1 - \frac{1}{e} - \varepsilon)C(S^*)$ with probability $\geq 1 - \delta$.

We may observe that the main consequence of Theorem 4.7 is that the number of samples required to obtain a $(1 - \frac{1}{e} - \varepsilon)$-approximation of S^* with probability $\geq 1 - \delta$ does not necessarily depend on the size of the graph G (e.g., the number of nodes $n = |V|$), but rather scales with the VC-dimension $VC(Q_k)$. Moreover, note that it always holds $VC(Q_k) \leq \log_2 |R_k| \leq \lfloor k \log_2(n) \rfloor$ (in other words, $VC(Q_k)$ cannot be larger than a naïve application of the union bound), but the gap between $VC(Q_k)$ and $\log_2 |R_k|$ (and $\lfloor k \log_2(n) \rfloor$) can be arbitrary large [63]. In order to characterize $VC(Q_k)$, we give an upper bound to it as a function of $VC(Q_1)$.

LEMMA 4.8. *The VC-dimension $VC(Q_k)$ of the range space Q_k is $VC(Q_k) \leq 2VC(Q_1)k \log_2(3k)$.*

Lemma 4.8 conveniently involves $VC(Q_1)$, a quantity that is often much easier to bound than $VC(Q_k)$. In fact, for the case of the betweenness centrality, it holds $VC(Q_1) \leq \lfloor \log_2(2B) \rfloor$, where B is (an upper bound to) the *vertex diameter* of the graph G [57], i.e., the maximum number of distinct nodes that are internal to a shortest path of G. Our next result generalizes such bound to general set centralities, in which we assume that the maximum number of distinct nodes in an hyper-edge h that we can sample from $\mathfrak{H}$ is upper bounded by a constant b.

LEMMA 4.9. *If $\sup_{h \in \mathfrak{H}} |h| \leq b$, then $VC(Q_1) \leq d_1 = \lfloor \log_2(2b) \rfloor$.*

For instance, when sampling simple paths of length at most $\kappa \geq 1$ for estimating the κ-path centrality it holds $b \leq \kappa$. For the triangle centrality, $b = 3$. Furthermore, note that $b \leq n$ as $h \subseteq V, \forall h \in \mathfrak{H}$. In light of Theorem 4.7 and Lemma 4.8, and under the assumption that $C(S^*) \in \Theta(1)$, as common in real-world networks [43, 67], it is immediate to observe that the sufficient sample size m to achieve an $(1 - \frac{1}{e} - \varepsilon)$-approximation is

$$m \in O\left(\frac{d_k + \log(1/\delta)}{\varepsilon^2}\right), m \in O\left(\frac{d_1 k \log(k) + \log(1/\delta)}{\varepsilon^2}\right), \tag{4}$$

where d_1 can be easily bounded with Lemma 4.9.

Remarkably, the bounds (4) are never worse than (1) (from the union bound, as $d_k \leq \lfloor k \log_2(n) \rfloor$), while they are significantly refined in many realistic scenarios. An interesting example is given by *small-world* networks [66], that have vertex diameter $B \in O(\log n)$; in such cases, both (3) and (4) provide an *exponential improvement*

on the dependence on n for set betweenness centrality approximations (as $b \leq B \in O(\log n)$, and $d_1 \in O(\log \log n)$). Furthermore, Lemma 4.8 and 4.9 imply that, when d_1 or d_k are constants (not depending on n), the sample complexity bounds of (4) are completely *independent* of the graph size n. This is the case, for example, when estimating the κ-path centrality and κ is a small constant (as it is typically in applications), where $d_1 = \lfloor \log_2(\kappa) \rfloor$, or for undirected graphs with unique shortest paths between all pairs of vertices[2], where $d_1 = 3$ (see Lemma 2 of [57]). These observations confirm that in many situations standard techniques do not capture the correct underlying complexity of the set centrality approximation task, yielding loose guarantees.

We remark that, while these results improve state-of-the-art bounds from a theoretical perspective, they also have a practical impact as they can be embedded in the progressive sampling strategy of CentRA by setting an upper limit to the number of samples to consider.

5 EXPERIMENTAL EVALUATION

In this Section we present our experiments. The goals of our experimental evaluation are: 1) to test the practical impact of the number t of Monte Carlo trials for estimating the AMCERA, both in terms of running time and bound on the SD $D(\mathcal{F}, \mathcal{H})$; 2) to compare the data-dependent bounds computed by CentRA with state-of-the-art techniques; 3) to show the impact, both in terms of running time and number of random samples, that a finer characterization of the SD has to the task of centrality maximization.

Experimental Setup. We implemented CentRA in C++. The code and the scripts to reproduce all experiments are available at https://github.com/leonardopellegrina/CentRA. All the code was compiled with GCC 8 and run on a machine with 2.30 GHz Intel Xeon CPU, 512 GB of RAM, on Ubuntu 20.04.

In our experiments we focus on the set betweenness centrality, and remand testing other centralities to the full version of this work. To sample random shortest paths from $\mathfrak{H}$, the procedure sampleHEs chooses a starting node u and final node v uniformly at random, and then picks a random shortest paths from u to v, using the balanced bidirectional BFS [14] (in which two BFSs are expanded from both u and v). If v is not reachable from u, the procedure returns the empty set. We compare CentRA with HEDGE [43], the state-of-the-art method for approximate centrality maximization. As the implementation of HEDGE is not publicly available, we implemented it is as a modification of CentRA. Note that we do not compare with the exact approach (that computes centralities exactly), as HEDGE [43] already outperforms it on small graphs, and because the exact approach does not conclude in reasonable time on the large graphs we considered[3], and to [67] (due to some flaws in the analysis pointed out by [43]). Note that CentRA does not use the upper bounds proved in Section 4.3 to limit its number of samples, so to directly test its data-dependent bounds. However, due to space constraints, we defer to the Appendix (Section A.4) analogous comparisons with a variant of CentRA, that we call

---

[2]In road networks, this feature is often enforced [29].

[3]Computing the centralities of *individual nodes* on large graphs already requires significant computation (e.g., [14] reports that 1 week is required on a 40 cores cluster for graphs of size comparable to the ones we tested). In addition, it is infeasible to update these centralities at each of the k iterations of the greedy algorithm.
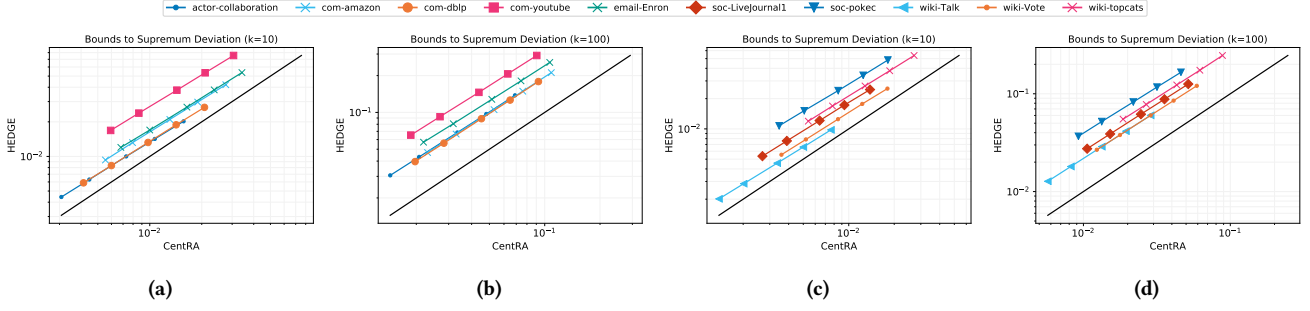
Figure 1: Comparison between the bounds to the Supremum Deviation $D(\mathcal{F}, \mathcal{H})$ **obtained by HEDGE (y axes, based on the union bound) and CENTRA (x axes, Section 4.2) on samples of size** $m \in \{5 \cdot 10^4, 10^5, 2 \cdot 10^5, 5 \cdot 10^5, 10^6\}$, **for** $k = 10$ **and** $k = 100$ **(other values in Figure 6). Figures (a)-(b): undirected graphs. Figures (c)-(d): directed graphs. Each point corresponds to a different value of m (the same for both algorithms). The black diagonal line corresponds to** $y = x$.
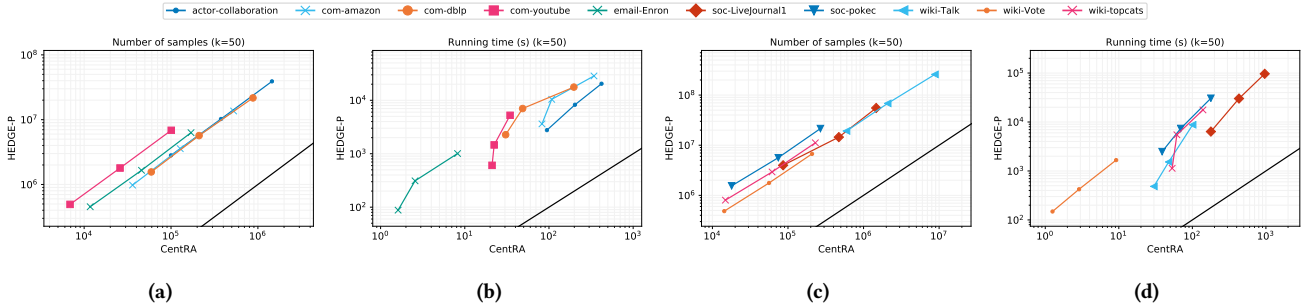
Figure 2: Comparison between the number of samples and running times (in seconds) to obtain an $(1 - 1/e - \varepsilon)$-**approximation of the optimal set centrality** $\mathcal{S}^*$ **using CENTRA (x axes) and HEDGE-P (y axes) for** $\varepsilon \in \{0.2, 0.1, 0.05\}$ **and** $k = 50$ **(other values of k shown in Appendix). Figures (a)-(b): samples and times for undirected graphs. Figures (c)-(d): directed graphs. Each point corresponds to a different value of ε (the same for both algorithms). The black diagonal line is at** $y = x$.

CENTRA-VC, that uses VC-dimension bounds (Section 4.3) instead of the MCERA (Section 4.1). We briefly comment these results in the following paragraphs. We repeat all experiments 10 times and report averages (standard deviations not shown since the experiments were extremely stable).

Graphs. We tested CENTRA on 5 undirected and 5 directed real-world graphs from SNAP[4] and KONECT[5]. The characteristics of the graphs are described in detail in Table 1 (in the Appendix).

Impact of CENTRA parameters. In our first experiments, we evaluate the practical impact of the number t of Monte Carlo trials used by CENTRA to derive data-dependent bounds to the SD via the AM-CERA (Section 4.1). We run CENTRA, letting it process a fixed number of samples $m = 5 \cdot 10^5$ using $t \in \{1, 10, 50, 10^2, 2.5 \cdot 10^2, 5 \cdot 10^2\}$, measuring the upper bound η to the SD (Theorem 4.4) and its running time. We focus on the two largest and two smallest graphs of the ones we considered, two undirected and two directed. In all cases, we fix $\delta = 0.05$ (note that we do not vary δ as its impact is negligible, due to the use of exponential tails bounds, see Thm. 4.4). In Figure 3 (a) we show the bounds to the SD, and in

Figure 3 (b) the running time (Figures in the Appendix). We conclude that, when using more than 10^2 trials, the improvements in terms of deviation bound is negligible. This confirms that a small number t of Monte Carlo trials is sufficient to estimate the ERA accurately. More importantly, the impact to the running time is negligible when increasing t on large graphs, and not predominant for smaller graphs (Figure 3 (b)). This means that the most expensive operations performed by CENTRA are sampling shortest paths and processing the sample with the greedy algorithm, while the AMCERA is computed efficiently. Based on these observations, we fix $t = 10^2$ for all experiments.

Bounds to the Supremum Deviation. We now compare the bounds to the Supremum Deviation $D(\mathcal{F}, \mathcal{H})$ obtained by HEDGE (using the union bound) with our analysis of Section 4.1 based on the AMCERA, at the core of CENTRA. In this way, we directly assess the trade-off in terms of accuracy and sample size m of both algorithms. To do so, we sample for both algorithms the same number m of hyper-edges and compute $\eta \geq D(\mathcal{F}, \mathcal{H})$ for CENTRA (using Thm. 4.4) and combine the Chernoff bound with the union bound (following the analysis of Lemma 2 of [43]) for HEDGE, obtaining $\eta^{UB} \geq D(\mathcal{F}, \mathcal{H})$. We considered $m \in \{5 \cdot 10^4, 10^5, 2 \cdot 10^5, 5 \cdot 10^5, 10^6\}$, $k \in \{10, 25, 50, 75, 100\}$, and $\delta = 0.05$ for both methods.

Figure 1 shows the results for these experiments. Figures 1 (a)-(b) show the bounds to the SD as functions of m for $k = 10$ and $k = 100$ for HEDGE (y axes) and CENTRA (x axes) for undirected graphs (to ease readability, directed are in Figures 1 (c)-(d)). Note that each point in the plots corresponds to a value of m, while the diagonal black line is at $y = x$. Figures for other values of k are very similar and showed in the Appendix (Figure 6) for space constraints. In Figure 4 we show the estimated centrality of the set S returned by CENTRA as function of k, for $m = 10^6$.

From these results, we can clearly see that CENTRA obtains a much smaller bound to the SD than HEDGE (as the points are above the black diagonal), uniformly for all graphs. This is a consequence of the sharp *data-dependent* bounds derived in Section 4.1, which significantly improve the loose guarantees given by the union bound. We observe that, for $k = 10$, the upper bound η^{UB} to the SD from the union bound is at least 1.25 times higher than η for all graphs, more than 1.5 higher for 7 graphs, and at least 2 times higher for 3 graphs. At $k = 50$ (Figure 6), it is 1.5 higher for all graphs, at lest 2 times higher for 6 graphs, and at least 3 times higher for 2 graphs. This gap is even larger for $k = 100$. Therefore, η is often *a fraction* of the upper bound η^{UB} from HEDGE. Interestingly, we observe this difference on both smaller and larger graphs. Furthermore, this gap grows with k, confirming that the data-dependent bounds of CENTRA scale much more nicely than the union bound w.r.t. k. To appreciate the magnitude of the improvement, note that both η and η^{UB} decrease proportionally with $\Theta(\sqrt{1/m})$ (see the rates in Figures 5); this means that to make the upper bound to the SD α times smaller, we need a sample that is approximately α^2 times larger. Therefore, we expect that obtaining a sharper bound to the SD should have a significant impact on the the sample size and, consequently, on the cost to analyze it. The following experiments confirm this intuition. Finally, the results deferred to Section A.4 (comparing CENTRA, CENTRA-VC, and HEDGE) confirm our theoretical insights: the novel VC-dimension bounds (used by CENTRA-VC) are always equally or more accurate than standard bounds, while CENTRA is the overall superior approach, as it leverages sharp data-dependent techniques.

Approximate Centrality Maximization with Progressive Sampling. In the last set of experiments, our goal is to evaluate the performance of CENTRA to compute an $(1 - 1/e - \varepsilon)$-approximation of the optimum S^* with progressive sampling. Note that HEDGE cannot be applied to this problem as is, since it requires knowledge of $C(S^*)$ to fix its sample size. Since CENTRA is the *first* algorithm of this kind, we compare it to a *new* variant of HEDGE, that we call HEDGE-P, that we implemented as a modification of CENTRA. HEDGE-P follows the same iterative scheme of Algorithm 1, but with crucial differences: instead of using the AMCERA to upper bound the SD, it uses η^{UB} (described previously) from a union bound (Lemma 2 of [43]). As the stopping condition, we follow the analysis of [43]: HEDGE-P stops when it guarantees that $D(\mathcal{F}, \mathcal{H}) \le \varepsilon C(S^*)/2 = \eta^{UB}$, as this is a sufficient condition to obtain an $(1 - 1/e - \varepsilon)$-approximation (see Thm. 1 of [43]). Instead, CENTRA leverages a refined stopping condition (line 14 of Alg. 1, Section 4.2). We compare CENTRA to HEDGE-P in terms of the number of samples and running times required to converge, using

$\varepsilon \in \{0.2, 0.1, 0.05\}$ and $k \in \{10, 50, 100\}$ (that are well representative of other values of k, as shown previously).

Figure 2 shows the results of these experiments for $k = 50$ (other values in Figure 7). CENTRA obtains an $(1 - 1/e - \varepsilon)$-approximation of the optimum S^* using a *fraction* of the samples and at a fraction of the time of HEDGE-P. For $k = 10$, CENTRA needs a random sample that is one order of magnitude smaller than HEDGE-P. For $k = 50$, the random samples required by CENTRA are at least 20 times smaller than the samples needed by HEDGE-P. When $k = 100$, the sample size reduction is close to 2 orders of magnitude (Figure 7). From these results it is clear that a better characterization of the SD and a refined stopping condition allow CENTRA to be a much more efficient algorithm for approximate centrality maximization in terms of sample sizes. As expected, we observed that reducing the sample sizes has a significant impact on the running times.

In fact, we observed the running time to be essentially linear w.r.t. the sample size. For $k = 10$, for 3 graphs and $\varepsilon = 0.2$, CENTRA is at least 5 times faster than HEDGE-P; for all other cases, CENTRA is one order of magnitude faster. For $k = 50$, CENTRA finished in less than 1/20 of the time needed by HEDGE-P, improving up to two orders of magnitude when $k = 100$ (Figure 7). Both CENTRA and HEDGE-P conclude after a small number of iterations (always at most 3), confirming that the adaptive progressive sampling schedule is very accurate. These observations confirm the efficiency of CENTRA: our new techniques provide efficiently computable, sharp accuracy bounds that enable much more scalable approximations for the centrality maximization task.

6 CONCLUSIONS

In this work we presented CENTRA, a new algorithm for approximate centrality maximization of sets of nodes. First, we developed a new approach based on efficiently computable bounds to Monte Carlo Rademacher Averages, a fundamental tool from statistical learning theory to obtain tight data-dependent bounds to the Supremum Deviation. Then, we derived new sample complexity bounds, proving that standard techniques typically provide overly conservative guarantees. We tested CENTRA on large real-world networks, showing that it significantly outperforms the state-of-the-art.

For future work, CENTRA can be extended to analyse dynamic [9, 10, 34], uncertain [60], and temporal networks [61], all settings in which our contributions may be useful to design efficient approximation algorithms for set centralities. Another direction is to consider the approximation of different group centralities that are not directly captured by the framework of Section 3.1 (e.g., [2, 8]). More generally, it would be interesting to adapt the techniques introduced in this work to other data mining problems, such as mining interesting [27, 59], significant [22, 32, 51–53], and causal patterns [64].

7 ACKNOWLEDGMENTS

REFERENCES

[1] Tharaka Alahakoon, Rahul Tripathi, Nicolas Kourtellis, Ramanuja Simha, and Adriana Iamnitchi. 2011. K-path centrality: A new centrality measure in social networks. In *Proceedings of the 4th workshop on social network systems*. 1–6.

[2] Eugenio Angriman, Ruben Becker, Gianlorenzo d'Angelo, Hugo Gilbert, Alexander van der Grinten, and Henning Meyerhenke. 2021. Group-Harmonic and Group-Closeness Maximization–Approximation and Engineering. In *2021 Proceedings of the Workshop on Algorithm Engineering and Experiments (ALENEX)*. SIAM, 154–168.

[3] Eugenio Angriman, Alexander van der Grinten, Aleksandar Bojchevski, Daniel Zügner, Stephan Günnemann, and Henning Meyerhenke. 2020. Group centrality maximization for large-scale graphs. In *2020 Proceedings of the Twenty-Second Workshop on Algorithm Engineering and Experiments (ALENEX)*. SIAM, 56–69.

[4] Martin Anthony, Peter L Bartlett, Peter L Bartlett, et al. 1999. *Neural network learning: Theoretical foundations*. Vol. 9. cambridge university press Cambridge.

[5] Peter L. Bartlett and Shahar Mendelson. 2002. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research* 3, Nov (2002), 463–482.

[6] Alex Bavelas. 1948. A mathematical model for group structures. *Human organization* 7, 3 (1948), 16–30.

[7] Elisabetta Bergamini, Michele Borassi, Pierluigi Crescenzi, Andrea Marino, and Henning Meyerhenke. 2019. Computing top-k closeness centrality faster in unweighted graphs. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 13, 5 (2019), 1–40.

[8] Elisabetta Bergamini, Tanya Gonser, and Henning Meyerhenke. 2018. Scaling up group closeness maximization. In *2018 Proceedings of the Twentieth Workshop on Algorithm Engineering and Experiments (ALENEX)*. SIAM, 209–222.

[9] Elisabetta Bergamini and Henning Meyerhenke. 2015. Fully-dynamic approximation of betweenness centrality. In *Algorithms-ESA 2015*. Springer, 155–166.

[10] Elisabetta Bergamini, Henning Meyerhenke, and Christian L Staudt. 2014. Approximating betweenness centrality in large evolving networks. In *2015 Proceedings of the Seventeenth Workshop on Algorithm Engineering and Experiments (ALENEX)*. SIAM, 133–146.

[11] Paolo Boldi and Sebastiano Vigna. 2013. In-core computation of geometric centralities with hyperball: A hundred billion nodes and beyond. In *2013 IEEE 13th International Conference on Data Mining Workshops*. IEEE, 621–628.

[12] Paolo Boldi and Sebastiano Vigna. 2014. Axioms for centrality. *Internet Mathematics* 10, 3-4 (2014), 222–262.

[13] Michele Borassi, Pierluigi Crescenzi, and Michel Habib. 2016. Into the square: On the complexity of some quadratic-time solvable problems. *Electronic Notes in Theoretical Computer Science* 322 (2016), 51–67.

[14] Michele Borassi and Emanuele Natale. 2019. KADABRA is an adaptive algorithm for betweenness via random approximation. *Journal of Experimental Algorithmics (JEA)* 24 (2019), 1–35.

[15] Christian Borgs, Michael Brautbar, Jennifer Chayes, and Brendan Lucier. 2014. Maximizing social influence in nearly optimal time. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*. SIAM, 946–957.

[16] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. 2013. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press.

[17] Olivier Bousquet. 2002. A Bennett concentration inequality and its application to suprema of empirical processes. *Comptes Rendus Mathematique* 334, 6 (2002), 495–500.

[18] Ulrik Brandes. 2001. A faster algorithm for betweenness centrality. *Journal of mathematical sociology* 25, 2 (2001), 163–177.

[19] Shiri Chechik, Edith Cohen, and Haim Kaplan. 2015. Average distance queries through weighted samples in graphs and metric spaces: high scalability with tight statistical guarantees.. In *APPROX/RANDOM*.

[20] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20 (1995), 273–297.

[21] Cyrus Cousins, Chloe Wohlgemuth, and Matteo Riondato. 2021. Bavarian: Betweenness Centrality Approximation with Variance-Aware Rademacher Averages. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 196–206.

[22] Sebastian Dalleiger and Jilles Vreeken. 2022. Discovering Significant Patterns under Sequential False Discovery Control. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 263–272.

[23] Alane M de Lima, Murilo VG da Silva, and André L Vignatti. 2020. Estimating the Percolation Centrality of Large Networks through Pseudo-dimension Theory. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1839–1847.

[24] Dóra Erdős, Vatche Ishakian, Azer Bestavros, and Evimaria Terzi. 2015. A divide-and-conquer algorithm for betweenness centrality. In *Proceedings of the 2015 SIAM International Conference on Data Mining*. SIAM, 433–441.

[25] Uriel Feige. 1998. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)* 45, 4 (1998), 634–652.

[26] Martin Fink and Joachim Spoerhase. 2011. Maximum betweenness centrality: approximability and tractable cases. In *WALCOM: Algorithms and Computation:*

5th International Workshop, WALCOM 2011, New Delhi, India, February 18-20, 2011. Proceedings 5. Springer, 9–20.

[27] Jonas Fischer and Jilles Vreeken. 2020. Discovering succinct pattern sets expressing co-occurrence and mutual exclusivity. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 813–823.

[28] Linton C Freeman. 1977. A set of measures of centrality based on betweenness. *Sociometry* (1977), 35–41.

[29] Robert Geisberger, Peter Sanders, and Dominik Schultes. 2008. Better approximation of betweenness centrality. In *2008 Proceedings of the Tenth Workshop on Algorithm Engineering and Experiments (ALENEX)*. SIAM, 90–100.

[30] Rumi Ghosh, Shang-hua Teng, Kristina Lerman, and Xiaoran Yan. 2014. The interplay between dynamics and networks: centrality, communities, and cheeger inequality. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1406–1415.

[31] Michelle Girvan and Mark EJ Newman. 2002. Community structure in social and biological networks. *Proceedings of the national academy of sciences* 99, 12 (2002), 7821–7826.

[32] Wilhelmiina Hämäläinen and Geoffrey I Webb. 2019. A tutorial on statistically sound pattern discovery. *Data Mining and Knowledge Discovery* 33 (2019), 325–377.

[33] Sariel Har-Peled and Micha Sharir. 2011. Relative (p, ϵ)-approximations in geometry. *Discrete & Computational Geometry* 45, 3 (2011), 462–496.

[34] Takanori Hayashi, Takuya Akiba, and Yuichi Yoshida. 2015. Fully dynamic betweenness centrality maintenance on massive networks. *Proceedings of the VLDB Endowment* 9, 2 (2015), 48–59.

[35] Vatche Ishakian, Dóra Erdős, Evimaria Terzi, and Azer Bestavros. 2012. A framework for the evaluation and management of network centrality. In *Proceedings of the 2012 SIAM International Conference on Data Mining*. SIAM, 427–438.

[36] Rishabh Iyer and Jeff Bilmes. 2012. Algorithms for Approximate Minimization of the Difference between Submodular Functions, with Applications. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence* (Catalina Island, CA) *(UAI'12)*. AUAI Press, Arlington, Virginia, USA, 407–417.

[37] Swami Iyer, Timothy Killingback, Bala Sundaram, and Zhen Wang. 2013. Attack robustness and centrality of complex networks. *PloS one* 8, 4 (2013), e59613.

[38] David Kempe, Jon Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. 137–146.

[39] Vladimir Koltchinskii. 2001. Rademacher penalties and structural risk minimization. *IEEE Transactions on Information Theory* 47, 5 (2001), 1902–1914.

[40] Vladimir Koltchinskii and Dmitriy Panchenko. 2000. Rademacher processes and bounding the risk of function learning. In *High dimensional probability II*. Springer, 443–457.

[41] Michel Ledoux and Michel Talagrand. 1991. *Probability in Banach Spaces: isoperimetry and processes*. Vol. 23. Springer Science & Business Media.

[42] Yi Li, Philip M Long, and Aravind Srinivasan. 2001. Improved bounds on the sample complexity of learning. *J. Comput. System Sci.* 62, 3 (2001), 516–527.

[43] Ahmad Mahmoody, Charalampos E Tsourakakis, and Eli Upfal. 2016. Scalable betweenness centrality maximization via sampling. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1765–1773.

[44] Pascal Massart. 2000. Some applications of concentration inequalities to statistics. *Annales de la Faculté des sciences de Toulouse: Mathématiques* 9, 2 (2000), 245–303.

[45] Sourav Medya, Arlei Silva, Ambuj Singh, Prithwish Basu, and Ananthram Swami. 2018. Group centrality maximization via network design. In *Proceedings of the 2018 SIAM International Conference on Data Mining*. SIAM, 126–134.

[46] Michael Mitzenmacher and Eli Upfal. 2017. *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press.

[47] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. 2018. *Foundations of machine learning*. MIT press.

[48] Mark Newman. 2018. *Networks*. Oxford university press.

[49] Mark EJ Newman. 2005. A measure of betweenness centrality based on random walks. *Social networks* 27, 1 (2005), 39–54.

[50] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. *The PageRank citation ranking: Bringing order to the web*. Technical Report. Stanford InfoLab.

[51] Leonardo Pellegrina, Cyrus Cousins, Fabio Vandin, and Matteo Riondato. 2022. MCRapper: Monte-Carlo Rademacher averages for poset families and approximate pattern mining. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 16, 6 (2022), 1–29.

[52] Leonardo Pellegrina, Matteo Riondato, and Fabio Vandin. 2019. Hypothesis testing and statistically-sound pattern mining. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 3215–3216.

[53] Leonardo Pellegrina, Matteo Riondato, and Fabio Vandin. 2019. SPuManTE: Significant pattern mining with unconditional testing. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1528–1538.

[54] Leonardo Pellegrina and Fabio Vandin. 2021. SILVAN: Estimating Betweenness Centralities with Progressive Sampling and Non-uniform Rademacher Bounds. *arXiv preprint arXiv:2106.03462* (2021).

[55] David Pollard. 2012. *Convergence of stochastic processes.* Springer Science & Business Media.

[56] Foster Provost, David Jensen, and Tim Oates. 1999. Efficient progressive sampling. In *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining.* 23–32.

[57] Matteo Riondato and Evgenios M. Kornaropoulos. 2016. Fast approximation of betweenness centrality through sampling. *Data Mining and Knowledge Discovery* 30, 2 (2016), 438–475.

[58] Matteo Riondato and Eli Upfal. 2018. ABRA: Approximating Betweenness Centrality in Static and Dynamic Graphs with Rademacher Averages. *ACM Trans. Knowl. Disc. from Data* 12, 5 (2018), 61.

[59] Matteo Riondato and Fabio Vandin. 2020. MiSoSouP: Mining interesting subgroups with sampling and pseudodimension. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 14, 5 (2020), 1–31.

[60] Arkaprava Saha, Ruben Brokkelkamp, Yllka Velaj, Arijit Khan, and Francesco Bonchi. 2021. Shortest paths and centrality in uncertain networks. *Proceedings of the VLDB Endowment* 14, 7 (2021), 1188–1201.

[61] Diego Santoro and Ilie Sarpe. 2022. ONBRA: Rigorous Estimation of the Temporal Betweenness Centrality in Temporal Networks. In *Proceedings of the ACM Web Conference 2022.* 1579–1588.

[62] Alexander Schrijver et al. 2003. *Combinatorial optimization: polyhedra and efficiency.* Vol. 24. Springer.

[63] Shai Shalev-Shwartz and Shai Ben-David. 2014. *Understanding Machine Learning: From Theory to Algorithms.* Cambridge University Press.

[64] Dario Simionato and Fabio Vandin. 2022. Bounding the Family-Wise Error Rate in Local Causal Discovery using Rademacher Averages. In *Proceedings of Mach. Learn. Knowl. Disc. Databases - European Conf., ECML PKDD 2022.*

[65] V. N. Vapnik and A. Ya. Chervonenkis. 1971. On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities. *Theory of Probability & Its Applications* 16, 2 (1971), 264. https://doi.org/10.1137/1116025

[66] Duncan J Watts and Steven H Strogatz. 1998. Collective dynamics of 'small-world'networks. *nature* 393, 6684 (1998), 440–442.

[67] Yuichi Yoshida. 2014. Almost linear-time algorithms for adaptive betweenness centrality using hypergraph sketches. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining.* 1416–1425.

A APPENDIX

In this Appendix we provide the proofs for our main results and additional experimental results that were deferred due to space constraints.

A.1 Proofs of Section 4.1

First, we state some key technical tools. The first is a contraction inequality for Rademacher averages.

LEMMA A.1 (LEMMA 26.9 [63]). *Let $\mathcal{F}$ be a family of functions from a domain $\mathcal{X}$ to $\mathbb{R}$. Let $\phi : \mathbb{R} \to \mathbb{R}$ be a L-Lipschitz function, such that, for all $a, b \in \mathbb{R}$ it holds $|\phi(a) - \phi(b)| \le L|a - b|$. Define σ as a vector of m i.i.d. Rademacher random variables $\sigma = < \sigma_1, \ldots, \sigma_m >$. For any $X \in \mathcal{X}^m$ with $X = \{x_1, \ldots, x_m\}$, it holds*

$$\mathbb{E}_{\sigma}\left[\sup_{f \in \mathcal{F}} \sum_{i=1}^{m} \sigma_i \phi(f(x_i))\right] \le L\mathbb{E}_{\sigma}\left[\sup_{f \in \mathcal{F}}\left\{\sum_{i=1}^{m} \sigma_i f(x_i)\right\}\right].$$

We now state a concentration inequality for functions uniformly distributed on the binary hypercube.

THEOREM A.2 (THEOREM 5.3 [16]). *For $c > 0$, let a function $g : \{-1, 1\}^c \to \mathbb{R}$ and assume that X is uniformly distributed on $\{-1, 1\}^c$. Let $v > 0$ be such that*

$$\sum_{i=1}^{c} \left(g(x) - g(\overline{x}^i)\right)_+^2 \le v$$

for all $x = (x_1, \ldots, x_c) \in \{-1, 1\}^c$, where

$$\overline{x}^i = (x_1, \ldots, x_{i-1}, -x_i, x_{i+1}, \ldots, x_c)$$
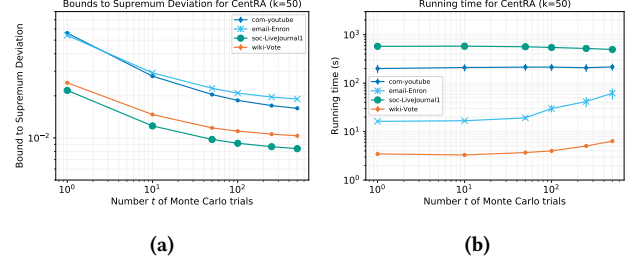

(a)　　　　　(b)

Figure 3: Impact of CENTRA parameters on upper bounds to the SD and running times on a sample of size $m = 5 \cdot 10^5$ for $k = 50$. (a): bounds to the SD as function of the number of Monte Carlo trials t. (b): running time of CENTRA.
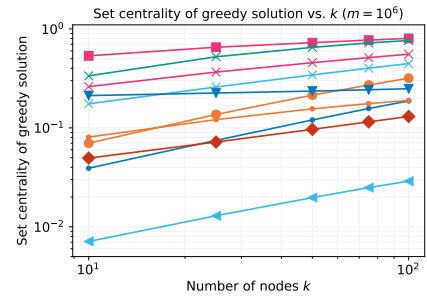

Figure 4: Estimated centrality $C_{\mathcal{H}}(S)$ of the node set S returned by CENTRA for different k and $m = 10^6$. (the legend is the same of Figure 1.)

Table 1: Statistics of undirected (top section) and directed (bottom section) graphs. B is the vertex diameter.

G	$|V|$	$|E|$	B
actor-collaboration	3.82e5	3.31e7	13
com-amazon	3.34e5	9.25e5	44
com-dblp	3.17e5	1.04e6	21
com-youtube	1.13e6	2.98e6	20
email-Enron	3.66e4	1.83e5	11
soc-LiveJournal1	4.84e6	6.90e7	16
soc-pokec	1.63e6	3.06e7	16
wiki-Talk	2.39e6	5.02e6	9
wiki-topcats	1.79e6	2.85e7	9
wiki-Vote	7.11e3	1.03e5	7

is a copy of x with the i-th component multiplied by -1, and $(b)_+ = \max\{b, 0\}$ is the positive part of $b \in \mathbb{R}$. Then, the random variable $Z \doteq g(X)$ satisfies, for all $q > 0$,

$$\Pr(Z > \mathbb{E}[Z] + q), \Pr(Z < \mathbb{E}[Z] - q) \le \exp(-q^2/v).$$

We are now ready to prove Theorem 4.1. To do so, we first prove that the expected AMCERA upper bounds the ERA with the contraction inequality of Lemma A.1, and then we show that the

AMCERA is sharply concentrated around its expectation, leveraging Theorem A.2.

PROOF OF THEOREM 4.1. We first prove that

$$\hat{R}(\mathcal{F}, \mathcal{H}) \le \mathbb{E}_\sigma \left[\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma) \right].$$

We observe that, for all $h \in \mathfrak{H}$ and all $\mathcal{S} \subseteq V$, we can equivalently write the function $f_\mathcal{S}$ as

$$f_\mathcal{S}(h) = \max \{ f_u(h) : u \in \mathcal{S} \} = \bigvee_{u \in \mathcal{S}} f_u(h)$$

$$= \min \left\{ 1, \sum_{u \in \mathcal{S}} f_u(h) \right\}.$$

We define the function $\phi : \mathbb{R} \to \mathbb{R}$ as

$$\phi(x) = \min\{1, x\}, \text{ for } x \in [0, k],$$

and observe that it is 1-Lipschitz. Replacing the above in the definition of the ERA $\hat{R}(\mathcal{F}, \mathcal{H})$, we have

$$\hat{R}(\mathcal{F}, \mathcal{H}) = \mathbb{E}_\sigma \left[\sup_{f_\mathcal{S} \in \mathcal{F}} \left\{ \frac{1}{m} \sum_{s=1}^m \sigma_s f_\mathcal{S}(h_s) \right\} \right]$$

$$= \mathbb{E}_\sigma \left[\sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \le k} \left\{ \frac{1}{m} \sum_{s=1}^m \sigma_s \phi \left(\sum_{u \in \mathcal{S}} f_u(h_s) \right) \right\} \right]. \quad (5)$$

We apply Lemma A.1 to upper bound (5) obtaining that

$$(5) \le \mathbb{E}_\sigma \left[\sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \le k} \left\{ \frac{1}{m} \sum_{s=1}^m \sigma_s \sum_{u \in \mathcal{S}} f_u(h_s) \right\} \right]$$

$$= \mathbb{E}_\sigma \left[\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma) \right].$$

We now derive a new concentration bound for $\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma)$ w.r.t. its expectation $\mathbb{E}_\sigma \left[\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma) \right]$. Let the matrix $x \in \{-1, 1\}^{t \times m}$. For $j \in [1, t]$ and $s \in [1, m]$, define the matrix $\overline{x}^{js}$ as a copy of x such that its (j, s)-th component $\overline{x}_{js}^{js}$ is equal to the (j, s)-th component x_{js} of x multiplied by -1, i.e. $\overline{x}_{js}^{js} = -x_{js}$. We define the function g as $g(x) = tm\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, x)$. We aim at upper bounding

$$\sum_{j=1}^t \sum_{s=1}^m \left(g(x) - g(\overline{x}^{js}) \right)_+^2$$

below some constant v. We first observe that the equivalence

$$\left(g(x) - g(\overline{x}^{js}) \right)_+ = \left(g(x) - g(\overline{x}^{js}) \right) \mathbb{1} \left[g(x) > g(\overline{x}^{js}) \right] \quad (6)$$

implies that we can focus on the case $g(x) > g(\overline{x}^{js})$ to upper bound (6), as otherwise (6) is equal to 0. Let S_j^* be the subset of V such that

$$S_j^* = \arg\max_{\mathcal{S} \subseteq V, |\mathcal{S}| \le k} \left\{ \sum_{u \in \mathcal{S}} \sum_{s=1}^m f_u(h_s) x_{js} \right\}.$$

We note that, from the definition of supremum and after simple calculations, for any $j \in [1, t]$ and $s \in [1, m]$ it holds

$$g(x) - g(\overline{x}^{js}) \le \sum_{u \in S_j^*} f_u(h_s) \left[x_{js} - \overline{x}_{js}^{js} \right]$$

$$= 2 \sum_{u \in S_j^*} f_u(h_s) x_{js},$$

since all entries of x and $\overline{x}^{js}$ are equal, apart from the pair (j, s) for which $x_{js} = -\overline{x}_{js}^{js}$. Summing over all j, s we obtain

$$\sum_{j=1}^t \sum_{s=1}^m \left(g(x) - g(\overline{x}^{js}) \right)_+^2 \le \sum_{j=1}^t \sum_{s=1}^m \left(2 \sum_{u \in S_j^*} f_u(h_s) x_{js} \right)^2$$

$$= 4 \sum_{j=1}^t \sum_{s=1}^m \left(\sum_{u \in S_j^* \cap h_s} f_u(h_s) \right)^2 \le 4 \sum_{j=1}^t \sum_{s=1}^m b_\mathcal{H} \sum_{u \in S_j^*} (f_u(h_s))^2$$

$$\le 4 b_\mathcal{H} t \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \le k} \left\{ \sum_{u \in \mathcal{S}} \sum_{s=1}^m (f_u(h_s))^2 \right\}$$

$$= 4 t m \tilde{w}_\mathcal{F}(\mathcal{H}),$$

where the second-last inequality holds by Cauchy-Schwarz inequality. We apply Theorem A.2 with $v = 4 t m \tilde{w}_\mathcal{F}(\mathcal{H})$, obtaining

$$\mathbb{E}_\sigma \left[tm\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma) \right] \le tm\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma) + q$$

with probability $\ge 1 - \exp(-q^2/v)$. Setting $\exp(-q^2/v) \le \delta$ and solving for q proves the statement. □

PROOF OF LEMMA 4.2. First, note that we have already shown

$$\hat{R}(\mathcal{F}_k, \mathcal{H}) \le \mathbb{E}_\sigma \left[\tilde{R}_m^t(\mathcal{F}_k, \mathcal{H}, \sigma) \right]$$

in the proof of Theorem 4.1. We now prove

$$\mathbb{E}_\sigma \left[\tilde{R}_m^t(\mathcal{F}_k, \mathcal{H}, \sigma) \right] \le k \hat{R}(\mathcal{F}_1, \mathcal{H}). \quad (7)$$

We have

$$\tilde{R}_m^t(\mathcal{F}_k, \mathcal{H}, \sigma) = \frac{1}{t} \sum_{j=1}^t \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \le k} \left\{ \sum_{u \in \mathcal{S}} \frac{1}{m} \sum_{s=1}^m \sigma_{js} f_u(h_s) \right\}$$

$$\le \frac{1}{t} \sum_{j=1}^t \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \le k} \left\{ \sum_{u \in \mathcal{S}} \sup_{u \in \mathcal{S}} \left\{ \frac{1}{m} \sum_{s=1}^m \sigma_{js} f_u(h_s) \right\} \right\}$$

$$\le \frac{k}{t} \sum_{j=1}^t \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \le k} \left\{ \sup_{u \in \mathcal{S}} \left\{ \frac{1}{m} \sum_{s=1}^m \sigma_{js} f_u(h_s) \right\} \right\}$$

$$= \frac{k}{t} \sum_{j=1}^t \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \le 1} \left\{ \sum_{u \in \mathcal{S}} \frac{1}{m} \sum_{s=1}^m \sigma_{js} f_u(h_s) \right\}$$

$$= k \tilde{R}_m^t(\mathcal{F}_1, \mathcal{H}, \sigma).$$

Taking the expectation w.r.t. σ on both sides yields (7). The rightmost inequality of the statement follows from simple properties of the supremum. □

We now show Theorem 4.3. Our proof leverages the concept of *self-bounding* functions [16]. Define the functions $g : \mathcal{X}^m \to \mathbb{R}$ and $g^i : \mathcal{X}^{m-1} \to \mathbb{R}$, let the vector $x = (x_1, \ldots, x_m) \in \mathcal{X}^m$ and define $x^i = (x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_m)$ as a copy of x removing the i-th entry x_i. We say that g is a self-bounding function if there exists a g^i such that, for all $x \in \mathcal{X}^m$, it holds

$$0 \le g(x) - g^i(x^i) \le 1,$$

and that

$$\sum_{i=1}^m (g(x) - g^i(x^i)) \le g(x).$$

If all the $x_i \in X$ are independent random variables, a self-bounding function satisfies (Theorem 6.12 [16])

$$\Pr\left(g(x) \leq \mathbb{E}_x[g(x)] - q\right) \leq \exp\left(-\frac{q^2}{2\mathbb{E}_x[g(x)]}\right).$$

PROOF OF THEOREM 4.3. For any sample $\mathcal{H}$ of size m, define $\mathcal{S}' = \arg\max_{\mathcal{S} \subseteq V, |\mathcal{S}| \leq k} C_{\mathcal{H}}(\mathcal{S})$. From Jensen's inequality it holds

$$C(\mathcal{S}^*) = \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \leq k} \mathbb{E}_{\mathcal{H}}[C_{\mathcal{H}}(\mathcal{S})]$$
$$\leq \mathbb{E}_{\mathcal{H}}\left[\sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \leq k} C_{\mathcal{H}}(\mathcal{S})\right] = \mathbb{E}_{\mathcal{H}}\left[C_{\mathcal{H}}(\mathcal{S}')\right].$$

We now show that the function $g : \mathfrak{H}^m \to \mathbb{R}$, with

$$g(\mathcal{H}) = mC_{\mathcal{H}}(\mathcal{S}') = \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \leq k} \sum_{s=1}^{m} f_{\mathcal{S}}(h_s)$$

is self-bounding. Define $g^i(\mathcal{H})$ as

$$g^i(\mathcal{H}) = \sup_{\mathcal{S} \subseteq V, |\mathcal{S}| \leq k} \sum_{s=1, s \neq i}^{m} f_{\mathcal{S}}(h_s).$$

First, denoting $x = \mathcal{H}$, note that $0 \leq g(x) - g^i(x^i) \leq f_{\mathcal{S}'}(h_i) \leq 1$ are immediate from properties of the supremum and the definition of f. Then, we have

$$\sum_{i=1}^{m}(g(x) - g^i(x^i)) \leq \sum_{i=1}^{m} f_{\mathcal{S}'}(h_i) = g(x).$$

Therefore, g is self-bounding, and it holds

$$\Pr\left(mC_{\mathcal{H}}(\mathcal{S}') \leq \mathbb{E}[mC_{\mathcal{H}}(\mathcal{S}')] - q\right) \leq \exp\left(\frac{-q^2}{2\mathbb{E}_{\mathcal{H}}[mC_{\mathcal{H}}(\mathcal{S}')]}\right).$$

Setting the r.h.s. $\leq \delta$, and solving for q, we obtain w.p. $\geq 1 - \delta$

$$\mathbb{E}[C_{\mathcal{H}}(\mathcal{S}')] \leq C_{\mathcal{H}}(\mathcal{S}') + \sqrt{\frac{2\mathbb{E}[C_{\mathcal{H}}(\mathcal{S}')]\ln(\frac{1}{\delta})}{m}}.$$

Finding the fixed point of the above inequality gives

$$\mathbb{E}[C_{\mathcal{H}}(\mathcal{S}')] \leq C_{\mathcal{H}}(\mathcal{S}') + \sqrt{\left(\frac{\ln(\frac{1}{\delta})}{m}\right)^2 + \frac{2C_{\mathcal{H}}(\mathcal{S}')\ln(\frac{1}{\delta})}{m}} + \frac{\ln(\frac{1}{\delta})}{m}. \tag{8}$$

Let $\mathcal{S}$ be the output of greedyCover$(k, \mathcal{H})$ for a fixed sample $\mathcal{H}$. For the properties of greedyCover, it holds

$$C_{\mathcal{H}}(\mathcal{S}') \leq C_{\mathcal{H}}(\mathcal{S})(1 - 1/e)^{-1}.$$

By replacing this upper bound to $C_{\mathcal{H}}(\mathcal{S}')$ in (8) and recalling that $C(\mathcal{S}^*) \leq \mathbb{E}[C_{\mathcal{H}}(\mathcal{S}')]$ we obtain the statement. $\qquad \square$

Before proving Theorem 4.4, we state some intermediate results. These results provide refined concentration inequalities relating Rademacher averages to the supremum deviation [16]. Let the *Rademacher complexity* $R(\mathcal{F}, m)$ of the set of functions $\mathcal{F}$ (defined in Section 3) be defined as $R(\mathcal{F}, m) = \mathbb{E}_{\mathcal{H}}\left[\hat{R}(\mathcal{F}, \mathcal{H})\right]$. The following central result relates $R(\mathcal{F}, m)$ to the *expected* supremum deviation.

LEMMA A.3. *Symmetrization Lemma [46, 63] Let Z be either*

$$\sup_{f \in \mathcal{F}} \{C_{\mathcal{H}}(\mathcal{S}) - C(\mathcal{S})\} \ \text{ or } \ \sup_{f \in \mathcal{F}} \{C(\mathcal{S}) - C_{\mathcal{H}}(\mathcal{S})\}.$$

It holds $\mathbb{E}_{\mathcal{H}}[Z] \leq 2R(\mathcal{F}, m)$.

The following theorem states Bousquet's inequality [17], that gives a variance-dependent bound to the supremum deviation above its expectation.

THEOREM A.4. *(Thm. 2.3 [17]) Let $\hat{v}_{\mathcal{F}} \geq \sup_{f \in \mathcal{F}}\{Var(f)\}$, and Z be either $\sup_{f \in \mathcal{F}}\{C_{\mathcal{H}}(\mathcal{S}) - C(\mathcal{S})\}$ or $\sup_{f \in \mathcal{F}}\{C(\mathcal{S}) - C_{\mathcal{H}}(\mathcal{S})\}$. Then, with probability at least $1 - \lambda$ over $\mathcal{H}$, it holds*

$$Z \leq \mathbb{E}_{\mathcal{H}}[Z] + \sqrt{\frac{2\ln(\frac{1}{\lambda})\left(\hat{v}_{\mathcal{F}} + 2\mathbb{E}[Z]\right)}{m}} + \frac{\ln(\frac{1}{\lambda})}{3m}.$$

The next result bounds $R(\mathcal{F}, m)$ above its estimate $\hat{R}(\mathcal{F}, \mathcal{H})$.

THEOREM A.5. *[16] With probability $\geq 1 - \lambda$ over $\mathcal{H}$, it holds*

$$R(\mathcal{F}, m) \leq \hat{R}(\mathcal{F}, \mathcal{H}) + \sqrt{\left(\frac{\ln(\frac{1}{\lambda})}{m}\right)^2 + \frac{2\ln\left(\frac{1}{\lambda}\right)\hat{R}(\mathcal{F}, \mathcal{H})}{m}} + \frac{\ln(\frac{1}{\lambda})}{m}.$$

We now prove Theorem 4.4.

PROOF OF THEOREM 4.4. We define 5 events $A_1, \ldots, A_5$ as

$$A_1 = \text{``}\sup_{f \in \mathcal{F}}\{C_{\mathcal{H}}(\mathcal{S}) - C(\mathcal{S})\} > \eta\text{''},$$
$$A_2 = \text{``}\sup_{f \in \mathcal{F}}\{C(\mathcal{S}) - C_{\mathcal{H}}(\mathcal{S})\} > \eta\text{''},$$
$$A_3 = \text{``}\hat{R}(\mathcal{F}, \mathcal{H}) > \tilde{R}\text{''},$$
$$A_4 = \text{``}R(\mathcal{F}, m) > R\text{''},$$
$$A_5 = \text{``}\sup_{f \in \mathcal{F}}\{C(\mathcal{S})\} > \nu\text{''}.$$

The statement holds if, with probability $\geq 1 - \delta$, A_1 and A_2 are both false. If we assume that $\Pr(A_i) \leq \delta/5, \forall i$, then it holds

$$\Pr\left(\bigcup_i A_i\right) \leq \sum_i \Pr(A_i) \leq \delta.$$

This implies that all events A_i are false simultaneously with probability $\geq 1 - \delta$, obtaining the statement. We now show that $\Pr(A_i) \leq \delta/5, \forall i$. We observe that $Pr(A_3) \leq \delta/5$ is a consequence of Theorem 4.1 (replacing δ by $\delta/5$). Then, $Pr(A_4) \leq \delta/5$ follows from Theorem A.5 (using $\lambda = \delta/5$ and $\hat{R}(\mathcal{F}, \mathcal{H}) \leq \tilde{R}$). $Pr(A_5) \leq \delta/5$ is a consequence of Theorem 4.3 (replacing δ by $\delta/5$). $Pr(A_1) \leq \delta/5$ and $Pr(A_2) \leq \delta/5$ both hold applying Bousquet's inequality (Theorem A.4) twice (with $\lambda = \delta/5$), observing that $R(\mathcal{F}, m) \leq R$, using the Symmetrization Lemma (Lemma A.3), and observing that

$$\sup_{f \in \mathcal{F}}\{Var(f)\} \leq \sup_{f \in \mathcal{F}}\{C(\mathcal{S})\} \leq \nu.$$

The fact $\Pr(A_i) \leq \delta/5, \forall i$, concludes our proof. $\qquad \square$

A.2 Proofs of Section 4.2

PROOF OF PROPOSITION 4.5. We prove that the statement is a consequence of the following facts, that we assume hold simultaneously with probability $\geq 1 - \delta/2^i$ at a fixed iteration $i \geq 0$:

(1) CENTRA computes $\tilde{R}_m^t(\mathcal{F}, \mathcal{H}, \sigma)$ correctly at line 11;

(2) η, at the end of iteration i, is computed using Theorem 4.4 such that $D(\mathcal{F}, \mathcal{H}_i) \leq \eta$;

(3) ξ (line 12) is such that $C(\mathcal{S}^*) \leq C_{\mathcal{H}_i}(\mathcal{S})/(1 - 1/e) + \xi$;

(4) $\mathcal{S}$ is computed by the greedy algorithm greedyCover$(k, \mathcal{H}_i)$ s.t. $C_{\mathcal{H}_i}(\mathcal{S}) \geq (1 - 1/e) \sup_{\mathcal{S}' \subseteq V, |\mathcal{S}'| \leq k} C_{\mathcal{H}_i}(\mathcal{S}')$;

(5) CentRA stops at the current iteration i when the stopping condition of line 14 is true.

Notice that if all the following inequalities hold,

$$C(\mathcal{S}^*) \leq \frac{C_{\mathcal{H}_i}(\mathcal{S})}{1 - \frac{1}{e}} + \xi \leq \frac{C_{\mathcal{H}_i}(\mathcal{S}) - \eta}{1 - \frac{1}{e} - \varepsilon} \leq \frac{C(\mathcal{S})}{1 - \frac{1}{e} - \varepsilon},$$

then $C(\mathcal{S}^*) \leq C(\mathcal{S})/(1 - 1/e - \varepsilon)$ and CentRA is correct. The rightmost inequality holds due to facts (1) and (2), while the leftmost holds from facts (3) and (4). The central inequality holds as it is equivalent to the stopping condition (line 14) and from fact (5).

We need to prove that the algorithm is correct with probability $\geq 1 - \delta$, taking into account the validity of the probabilistic guarantees for of all possible iterations i. The probability that there is at least one iteration i in which any of the above facts is false is at most $\sum_i \delta_i = \sum_i \delta 2^{-i} \leq \delta$, from a union bound, proving the statement. $\square$

PROOF OF LEMMA 4.6. When a sample $h \in \mathcal{H}$ is generated, $O(|h|t)$ operations are sufficient to generate t Rademacher random variables and update all values of r_j^u, for all $u \in h$ and $j \in [1, t]$. With t heaps, each of size at most n, the t sets of values $\{r_j^u : u \in V\}$ can be kept sorted with a total of $O(|h|t \log(n))$ time for each sample h. Summing such costs for all $h \in \mathcal{H}$ (noting that each sample is considered only once) and observing that $|h| \leq b_{\mathcal{H}}$, we obtain the first term of the bound. The second term is the time to retrieve the k largest positive values from all heaps, after processing all m_i samples, that costs $O(kt \log(n))$ operations. $\square$

A.3 Proofs of Section 4.3

Let $\mathcal{F}$ be a function family from a domain $\mathcal{X}$ to $\{0, 1\}$, and define the range space $Q = (\mathcal{X}, R)$ such that R is the family of subsets of $\mathcal{X}$ generated by $\mathcal{F}$:

$$R = \{\{x \in \mathcal{X} : f(x) = 1\} : f \in \mathcal{F}\}.$$

A sample $\mathcal{H} = \{h_1, \dots, h_m\} \in \mathcal{X}^m$ of size m taken i.i.d. from a distribution γ gives an (r, θ)-relative approximation for $\mathcal{F}$ if, for $r > 0$ and $\theta \in (0, 1]$, it holds

$$\left| \sum_{i=1}^{m} f(h_i) - \mathbb{E}_{h \sim \gamma}[f(h)] \right| \leq r \max\{\theta, \mathbb{E}_{h \sim \gamma}[f(h)]\}, \forall f \in \mathcal{F}.$$

We state a result due to [42] (in a version presented by [33]).

THEOREM A.6 (THM. 2.11 [33]). *Let* $d \geq VC(Q)$. *A random sample* $\mathcal{H}$ *of size* m *taken i.i.d. from a distribution* γ *gives an* (r, θ)-*relative approximation for* $\mathcal{F}$ *with probability* $\geq 1 - \delta$ *if*

$$m \geq c \frac{d \ln\left(\frac{1}{\theta}\right) + \ln\left(\frac{1}{\delta}\right)}{r^2 \theta},$$

where c *is an absolute constant.*

Note that the sample complexity bound of Theorem A.6 is optimal up to constant factors [42]. We now prove Theorem 4.7.

PROOF OF THEOREM 4.7. In accordance with the statement, assume m to be

$$m = 4c \frac{d_k \ln\left(\frac{1}{C(\mathcal{S}^*)}\right) + \ln\left(\frac{1}{\delta}\right)}{\varepsilon^2 C(\mathcal{S}^*)},$$

where c is the absolute constant of Theorem A.6. For $r = \varepsilon/2$, $\theta = C(\mathcal{S}^*)$, and $Q = Q_k$, the guarantees of Theorem A.6 imply that the sample $\mathcal{H}$ provides an $(\varepsilon/2, C(\mathcal{S}^*))$-relative approximation for $\mathcal{F}$ (where $\mathcal{F}$ is defined in Section 3.1) with probability $\geq 1 - \delta$. From the definition of $(\varepsilon/2, C(\mathcal{S}^*))$-relative approximation, it holds

$$D(\mathcal{F}, \mathcal{H}) \leq \varepsilon C(\mathcal{S}^*)/2.$$

Following analogous derivations of the proof of Theorem 1 in [43], this constraint to the SD is a sufficient condition to guarantee that $\mathcal{S}$ provides an $(1 - 1/e - \varepsilon)$-approximation of $\mathcal{S}^*$ with probability $\geq 1 - \delta$, proving the statement. $\square$

We now prove that the VC-dimension $VC(Q_k)$ of the range space Q_k can be upper bounded in terms of the VC-dimension $VC(Q_1)$ of the range space Q_1 (Lemma 4.8). We follow steps that are similar to the proof of bounds to the VC-dimension of a concept class composed by the intersection of up to k concept classes (Exercise 3.23 of [47]). In our case, R_k can be seen as the *disjunction* of up to k ranges from R_1.

PROOF OF LEMMA 4.8. We first prove that, for all $k \geq 1$,

$$|P(\mathcal{H}, R_k)| \leq |P(\mathcal{H}, R_1)||P(\mathcal{H}, R_{k-1})|,$$

noting that when $k = 1$ it holds $|P(\mathcal{H}, R_{k-1})| = 1$. Define the set of ranges R_k^v built from the set of nodes $V \setminus \{v\}$ as

$$R_k^v = \{\{h : f_{\mathcal{S}}(h) = 1\} : \mathcal{S} \subseteq V \setminus \{v\}, |\mathcal{S}| \leq k\}.$$

Clearly, it holds $R_k^v \subseteq R_k, \forall v \in V$. We can write R_k as the union of the ranges of individual nodes v with the ranges of R_{k-1}^v:

$$R_k = \{\{h : f_{\mathcal{S}}(h) = 1\} : \mathcal{S} \subseteq V, |\mathcal{S}| \leq k\}$$

$$= \bigcup_{v \in V} \{\{h : f_v(h) = 1\} \cup \{h : f_{\mathcal{S}}(h) = 1\} : \mathcal{S} \subseteq V \setminus \{v\}, |\mathcal{S}| \leq k - 1\}$$

$$= \bigcup_{v \in V} \{\{h : f_v(h) = 1\} \cup r : r \in R_{k-1}^v\}.$$

Therefore, since $R_{k-1}^v \subseteq R_{k-1}$ for all v, the set of projection $P(\mathcal{H}, R_k)$ of R_k on $\mathcal{H}$ is contained in the union of all the projections of $P(\mathcal{H}, R_{k-1})$ in conjunction with all projections $x \in P(\mathcal{H}, R_1)$:

$$P(\mathcal{H}, R_k) \subseteq \bigcup_{x \in P(\mathcal{H}, R_1)} \{(\mathcal{H} \cap r) \cup x : r \in R_{k-1}\}.$$

This implies that the size $|P(\mathcal{H}, R_k)|$ of $P(\mathcal{H}, R_k)$ can be bounded with an union bound by

$$|P(\mathcal{H}, R_k)| \leq \sum_{x \in P(\mathcal{H}, R_1)} |\{(\mathcal{H} \cap r) \cup x : r \in R_{k-1}\}|$$

$$\leq \sum_{x \in P(\mathcal{H}, R_1)} |\{\mathcal{H} \cap r : r \in R_{k-1}\}|$$

$$= \sum_{x \in P(\mathcal{H}, R_1)} |P(\mathcal{H}, R_{k-1})|$$

$$= |P(\mathcal{H}, R_1)||P(\mathcal{H}, R_{k-1})|,$$

where the second inequality holds since the number of intersections of the ranges R_{k-1} on the sample $\mathcal{H}$ can only decrease when in conjunctions with a fixed subset $x \subseteq \mathcal{H}$.

Iterating this argument k times, we obtain

$$|P(\mathcal{H}, R_k)| \leq |P(\mathcal{H}, R_1)|^k.$$

To prove that the VC-dimension of the range space Q_k is $< m$, it must hold that

$$|P(\mathcal{H}, R_1)|^k < 2^m.$$

Let $d = VC(Q_1)$. We apply Sauer-Shelah's Lemma [63] obtaining

$$|P(\mathcal{H}, R_1)|^k \leq \left(\frac{em}{d}\right)^{kd}.$$

We now need to show that, for $m = 2dk \log_2(3k)$, it holds

$$\left(\frac{em}{d}\right)^{kd} < 2^m.$$

Taking the $\log_2$ on both sides, and substituting $m = 2dk \log_2(3k)$, we obtain

$$\left(\frac{em}{d}\right)^{kd} < 2^m$$
$$\iff \log_2(2ek \log_2(3k)) < \log_2(9k^2)$$
$$\iff 2ek \log_2(3k) < 9k^2$$
$$\iff \log_2(3k) < \frac{9k}{2e},$$

which holds for all $k \geq 1$, proving the statement. $\qquad\square$

PROOF OF LEMMA 4.9. Denote a set $\mathcal{H} = \{h_1, \ldots, h_m\}$ of size m that is shattered, such that

$$|P(\mathcal{H}, R_1)| = 2^m.$$

If $\mathcal{H}$ is shattered, it means that each $h_i \in \mathcal{H}$ is in the projection of 2^{m-1} distinct ranges; there must be at least 2^{m-1} distinct nodes

$u \in V$ with $f_u(h_i) = 1$. This implies that $|h_i| \geq 2^{m-1}$. Since $|h| \leq b, \forall h \in \mathfrak{H}$, it also holds that

$$2^{m-1} \leq b \iff m \leq \log_2(2b).$$

Therefore, the maximum size of a set that can be shattered is at most $\lfloor \log_2(2b) \rfloor$, obtaining the statement. $\qquad\square$

A.4 Additional Experimental Results

In this Section we report additional results not included in the main text due to space constraints.

Comparisons between HEDGE, CentRA, and VC-dimension bounds. Figures 8 and 9 compare the bounds to the Supremum Deviation obtained from the results described in Section 4.3 (based on the VC-dimension), with results obtained by HEDGE (based on the union bound) and CentRA (based on Monte Carlo Rademacher Averages, Section 4.1). To ease the comparison, we denote a modification of CentRA, that we call CentRA-VC, that instead of using Rademacher Averages (the results of Section 4.1) uses the VC-dimension based bounds (from Section 4.3). For these experiments we consider $\delta = 0.05$, $k \in \{10, 50, 100\}$, and samples of size $m \in \{5 \cdot 10^4, 10^5, 2 \cdot 10^5, 5 \cdot 10^5, 10^6\}$ (other values of k produced analogous results). The bounds of Section 4.3 are computed bounding the VC-dimension $V(Q_k)$ with Lemma 4.8 and Lemma 4.9, using $b = B$ (the vertex diameter of the graphs, see Table 1).

From Figure 8, comparing CentRA-VC with HEDGE, we observe that CentRA-VC yields generally more accurate bounds, up to 4 times smaller for two graphs, confirming that standard techniques provide too conservative guarantees in such cases.

Figure 9 compares CentRA-VC with CentRA. As we may have expected, we can clearly conclude that distribution- and data-dependent bounds are significantly more accurate that VC-dimension based results in almost all cases, improving up to a factor 3. This confirms the significance of the contributions at the core of CentRA.
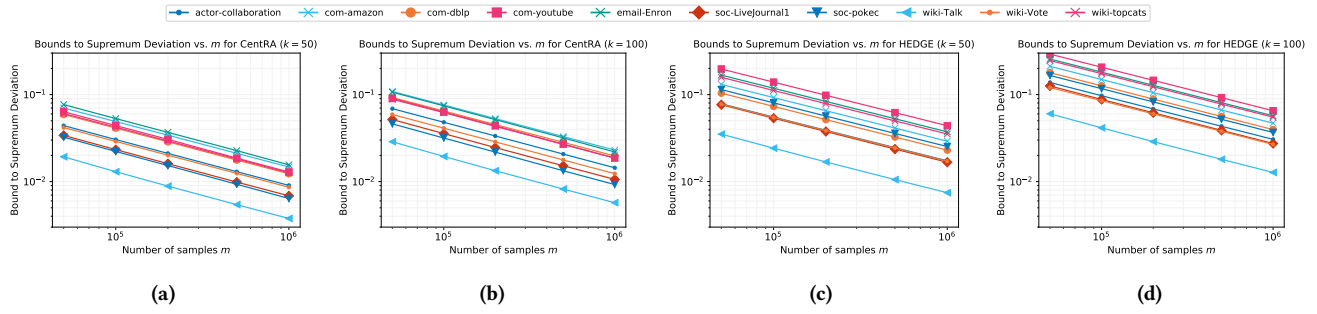
Figure 5: Bounds to the SD using CENTRA (Figures (a)-(b)) and HEDGE (Figures (c)-(d)) as functions of m, for $k = 50$ and $k = 100$.
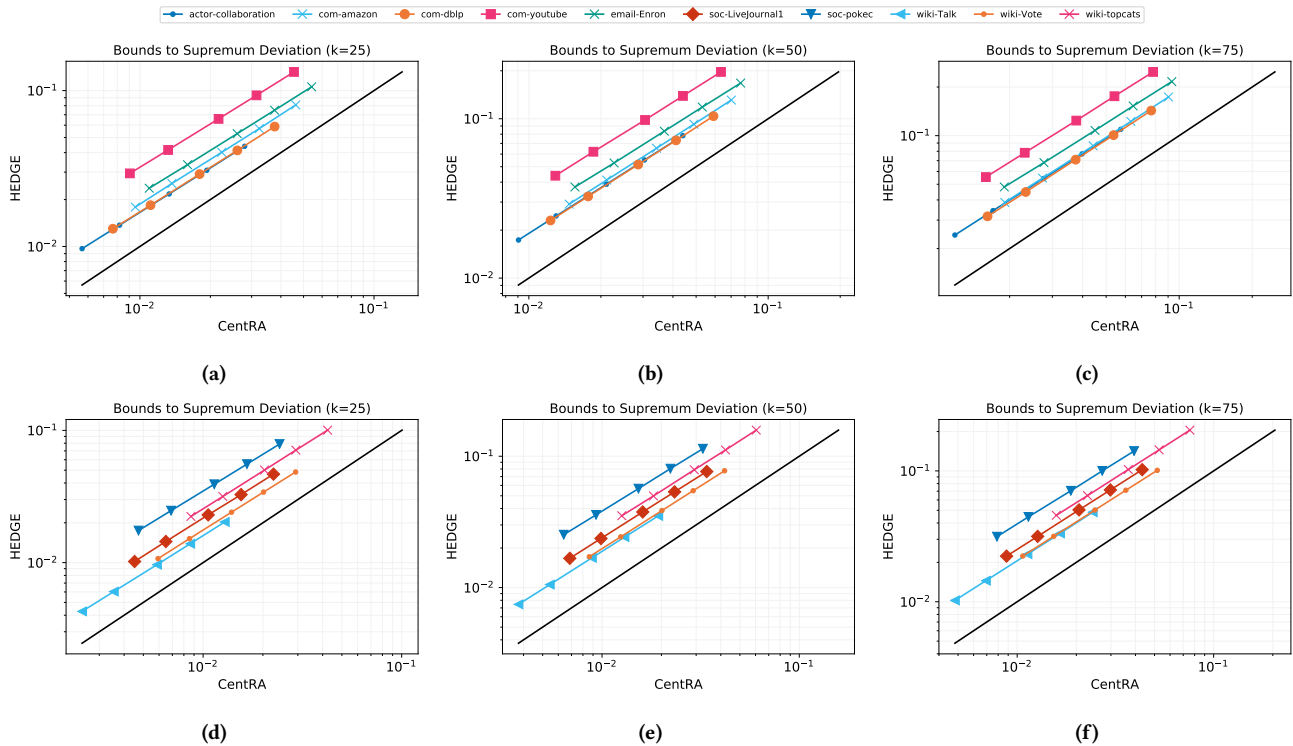

Figure 6: Additional Figures comparing the bounds to the Supremum Deviation $D(\mathcal{F}, \mathcal{H})$ obtained by HEDGE (y axes, based on the union bound) and CENTRA (x axes, Section 4.2) on samples of size m, for $k \in \{25, 50, 75\}$. Figures (a)-(c): undirected graphs. Figures (d)-(f): directed graphs. Each point corresponds to a different value of m. The black diagonal line corresponds to $y = x$.
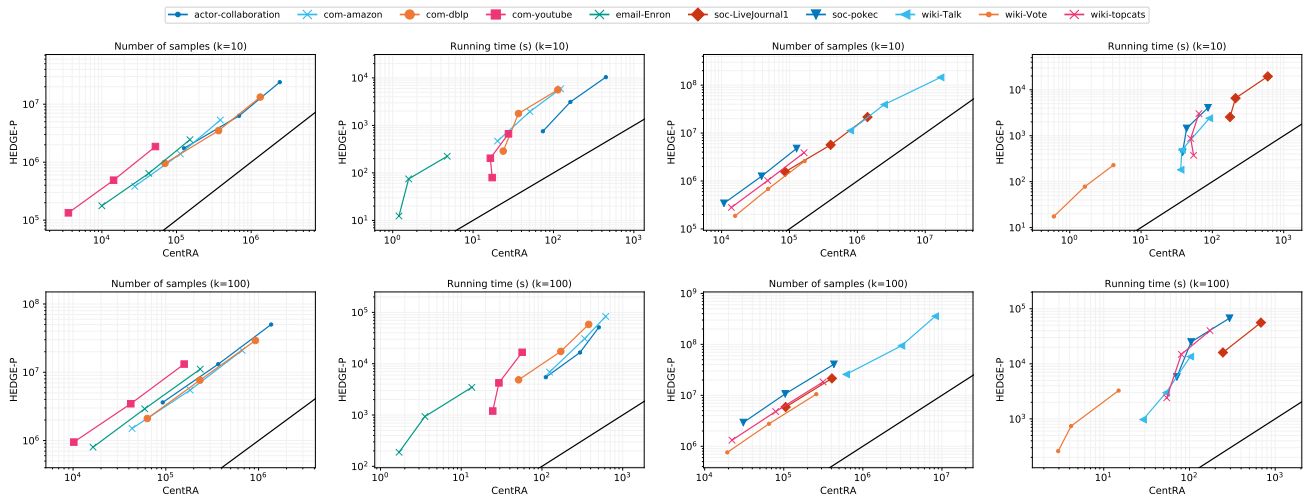
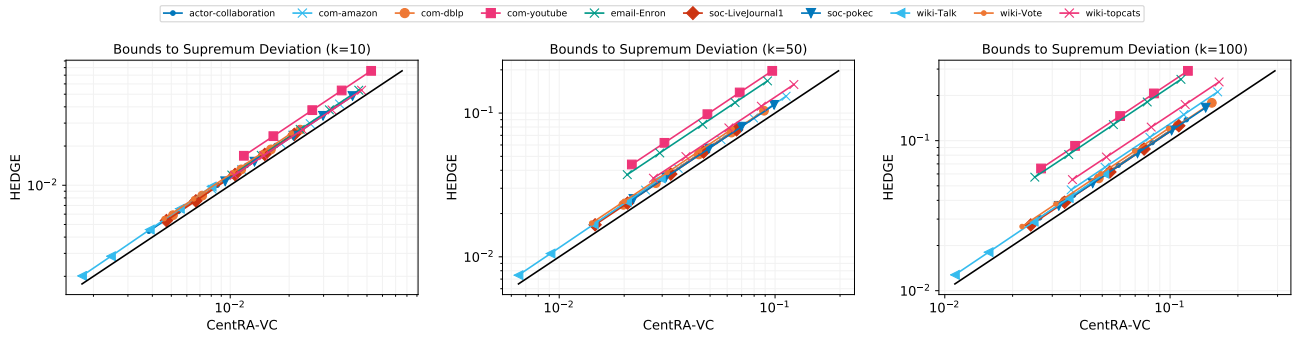Figure 7: Additional results (analogous to Figure 2) for $k = 10$ and $k = 100$.

Figure 8: Comparison between the bounds to the Supremum Deviation $\mathrm{D}(\mathcal{F}, \mathcal{H})$ obtained by HEDGE (y axes, based on the union bound) and CentRA-VC (x axes, using the VC-dimension bounds of Section 4.3) on samples of size $m \in \{5 \cdot 10^4, 10^5, 2 \cdot 10^5, 5 \cdot 10^5, 10^6\}$, for $k \in \{10, 50, 100\}$. Each point corresponds to a different value of m. The black diagonal line is at to $y = x$.
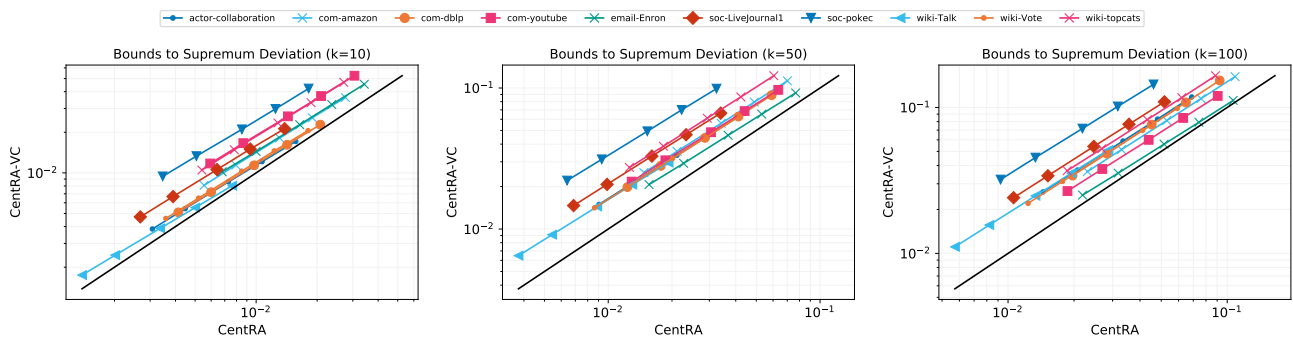

Figure 9: Comparison between the bounds to the Supremum Deviation $\mathrm{D}(\mathcal{F}, \mathcal{H})$ obtained by CentRA-VC (y axes, using the VC-dimension bounds of Section 4.3) and CentRA (x axes, using the Rademacher averages bounds of Section 4.1) on samples of size $m \in \{5 \cdot 10^4, 10^5, 2 \cdot 10^5, 5 \cdot 10^5, 10^6\}$, for $k \in \{10, 50, 100\}$. Each point corresponds to a different value of m. The black diagonal line is at to $y = x$.