An Extensive Characterization of Graph Sampling Algorithms

S. Haleh S. Dizaji
University of Klagenfurt
Klagenfurt am Wörthersee, Austria
Seyedehhaleh.Seyeddizaji@aau.at

Jože M. Rožanec
Jožef Stefan Institute
Ljubljana, Slovenia
joze.rozanec@ijs.si

Reza Farahani
University of Klagenfurt
Klagenfurt am Wörthersee, Austria
reza.farahani@aau.at

Dumitru Roman
SINTEF
Oslo, Norway
dumitru.roman@sintef.no

Radu Prodan
University of Klagenfurt
Klagenfurt am Wörthersee, Austria
radu.prodan@aau.at

ABSTRACT

While graph sampling is key to scalable processing, little research has tried to thoroughly compare and understand how it preserves features such as degree, clustering, and distances dependent on the graph size and structural properties. This research evaluates twelve widely adopted sampling algorithms across synthetic and real datasets to assess their qualities in three metrics: degree, clustering coefficient (CC), and hop plots. We find the random jump algorithm to be an appropriate choice regarding degree and hop-plot metrics and the random node for CC metric. In addition, we interpret the algorithms' sample quality by conducting correlation analysis with diverse graph properties. We discover eigenvector centrality and path-related features as essential features for these algorithms' degree quality estimation, node numbers (or the size of the largest connected component) as informative features for CC quality estimation and degree entropy, edge betweenness and path-related features as meaningful features for hop-plot metric. Furthermore, with increasing graph size, most sampling algorithms produce better-quality samples under degree and hop-plot metrics.

CCS CONCEPTS

• **Applied computing** → *Computer forensics*; *System forensics*;

KEYWORDS

Graph sampling algorithms, Scalable graph processing

ACM Reference Format:
S. Haleh S. Dizaji, Jože M. Rožanec, Reza Farahani, Dumitru Roman, and Radu Prodan. 2024. An Extensive Characterization of Graph Sampling Algorithms. In *Companion of the 15th ACM/SPEC International Conference on Performance Engineering (ICPE '24 Companion), May 7–11, 2024, London, United Kingdom.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3629527.3652899

ICPE '24 Companion, May 7–11, 2024, London, United Kingdom
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0445-1/24/05
https://doi.org/10.1145/3629527.3652899

1 INTRODUCTION

Graphs offer a flexible approach to modeling connected components and carry useful information about relationships of the structured data. However, accessing or processing full graphs in large-scale scenarios is infeasible or poses considerable challenges. For example, computing measures such as shortest paths, clusterings, or betweenness centrality (BC) become impractical [12] on large graphs. In such scenarios, *graph sampling* [12] is a popular remedy that allows for estimating these properties from a small fraction of its nodes and edges [25]. In addition, sampling can benefit machine learning tasks, with training more effectively on smaller fractions of the data. In particular, it can directly influence the robustness [3] and performance [1] of graph neural networks.

As the graph sampling algorithms become more extensive, studying their behavior becomes more demanding, as they perform differently depending on desired quality metrics and graphs. Unfortunately, literature remains scarce, and few works address this area, considering the limited amount of synthetic or real graphs. Furthermore, they do not provide an in-depth analysis of sampling quality considering graph size and structural features.

To bridge this void, we compare twelve graph sampling methods across around 2900 synthetic graphs of six types and twelve real datasets. We assess them using three metrics considered in the literature [12, 27], i.e., degree, clustering coefficient (CC), and hop-plots, to evaluate the qualities of samples regarding the original graphs. We quantify the dependency of these properties on graph features (77 features) and find the most relevant ones for each algorithm and metric. We uncover some important dependencies and highlight the most relevant features for different algorithms regarding each metric. In addition, we evaluate algorithms on small and large real graphs, confirming some of the relevant features obtained for synthetic ones.

The paper has seven sections. Section 2 reviews relevant sampling algorithms. Section 3 introduces related studies to our research. Section 4 defines the metrics used for evaluating samplings' result quality. Section 5 explains the experimental setup, including datasets, and experimental settings. Section 6 analyzes the results. Finally, section 7 concludes the paper and outlines future research.

2 GRAPH SAMPLING ALGORITHMS

We characterize graph sampling algorithms for static networks under three categories: node, edge, and traversal-based sampling [12]. This paper contributes to the state-of-the-art by investigating the sampling qualities of twelve popular algorithms of the three categories under various graph properties.

2.1 Node-based sampling

Node-based methods are most intuitive but only weakly preserve properties of specific graph types [2, 22], possibly losing connectivity [9]. *Random node (RN)* can preserve the CC for some graphs [9] and the degree distribution for random graphs [22], however poorly preserves the power-law degree distribution [14, 22] and average path length (APL) for non-small samples. *Random degree node (RDN)* applies probabilistic node selection proportional to the degrees [12], but loses degree distribution by creating bias over high-degree nodes [22]. *Random PageRank node* mitigates this bias [14] using nodes PageRank scores [20]. *Node sampling with contraction* reduces the graph's size by randomly removing nodes [6].

2.2 Edge-based sampling

Edge-based sampling can preserve edge-dependent properties, such as path length [2]. On the other hand, primary edge-based samplers have bias over high-degree nodes and poorly preserve some properties, such as connectivity and clustering. *Random edge (RE)* has poor preservation of graph structure (higher APLs for larger samples and lower CC). *Random node edge (RNE)* randomly selects a node and its edge [12]. RN selection mitigates bias over high-degree nodes [12]; however it can generate sparse graphs [14] due to limited edge selection. To solve this problem, *hybrid sampling* performs RNE or RE steps probabilistically [12], resulting in less bias towards high-degree nodes than RE. *Induced random edge (IRE)*, an extension of RE, performs an induction step by adding all edges between selected nodes in RE, collecting more information and better preserving the topological properties [2]. *Edge sampling with contraction* generates samples by randomly removing an edge and merging nodes previously joined by that edge [6].

2.3 Traversal-based sampling

Traversal-based methods improve the performance of RN and RE methods by capturing topological information of graph [2, 6].

Random traversal methods. Random walk (RW) performs sampling initialized from one seed node [21] with a better degree distribution estimation [18], but can get stuck in a graph region. To overcome this problem, *random jump (RJ)* jumps to a random node with some probability. *Metropolis-Hastings random walk (MHRW)* selects the neighboring nodes in RW proportional to degree ratios [23], but fails to estimate the degree distribution well [18]. *Multiple independent random walkers* avoid sampling from a specific region [6], [4], resulting in higher estimation errors [17].

Neighborhood exploration methods. Snowball (SB) traverses the graph by selecting a fixed number of neighbors of the current node set [5, 6], which preserves CC for certain graphs [9], but suffers from boundary bias [9], underestimating power-low degree distribution exponent and lower APL [9]. *Forest fire (FF)* adapted from the evolution network model [10] mitigates the local sampling problem of SB with the neighborhood size following a geometric distribution [6] with a bias over high-degree nodes and getting stuck in isolated clusters regions [14]. *Frontier sampling (FS)* performs probabilistic node selection from the current set according to its degree and replaces it randomly with one of its neighbors [17]; however, increasing the number of seed nodes (infinitely) results in uniform node and edge distribution [6]. *Expansion sampling (XS)* aims to preserve some graph community structure [13, 27] by starting from a random seed and traversing the neighborhood by selecting the node maximizing out-links of the current sample. *Rank degree (RD)* preserves community structure [27] by ranking the node neighborhood by degrees [24], randomly selecting a node from a seed set and its top-k neighbors as sample edges and replacing the seed set with them. *Tight sampling (TS)* mitigates the local sampling of SB trying to preserve local clusters around seed nodes [8]. *List sampling (LS)* tries to solve poor neighborhood exploration using a list of currently sampled nodes' neighbors [28] and has a better APL estimation on graphs with high CC [27].

3 RELATED WORK

We summarize the studies for graph sampling algorithms analysis in two sections: analytical and numerical evaluations.

3.1 Analytical evaluations

Stumpf and Wiu [22] analyzed RN on random, exponential and scale-free graphs and Lee et al. [9] studied RN and RE on Albert-Barabasi (AB) and real graphs. They characterized the degree distribution of samples dependent on the original graph degree distribution and sampling rate. Illenberger and Flötteröd [7] analyzed SB algorithm on Erdos-Renyi (ER) and real graphs and concluded that the original graphs' mean degree, degree correlation, and CC estimation quality decrease with the increasing variance of the original graph degree distribution. Ribeiro and Towsley [18] analyzed RN, RE, RW, and MHRW, estimating the graph degree distribution based on the unbiased Horvitz-Thompson estimator dependent on sample degrees and distributions and verified on large real graphs.

Limitations. While providing accurate estimations, these analyses study limited sampling algorithms and synthetic graphs and do not consider various graph properties. We analyze several algorithms (including updated algorithms) under six synthetic and twelve real graphs, considering several graph features.

3.2 Numerical evaluation

Leskovec and Faloutsos [12] evaluated ten node, edge, and traversal-based algorithms under scale-down and back-in-time samplings using nine metrics (i.e., degree, CC, connected components sizes, hop-plots, and singular values distributions) over four real graphs concluding that traversal-based algorithms yield better results for static graphs. Yoon et al. [26] evaluated RW under quality metrics, i.e., degree distribution, CC, and degree-degree correlation for Albert-Barabasi (AB) and three real graphs and found for high power-law degree distribution exponents, RW preserves most topological properties and reported deviations in small samples' degree distribution exponents with increasing the exponent. Lee et al. [9] studied RN, RE, and SB under degree, BC, APL, assortativity, and CC and found very different quantities of these properties for these algorithms. Zhang et al. [29] studied fourteen samplers of all categories using random and real graphs under numerical quality metrics (degree, BC, and hop-plots distributions), visualization, and execution time and discovered that the algorithm's performance depends on graph type, size, and measured property. Yousuf et al. [27] evaluated five traversal-based algorithms for twelve large real and three synthetic graphs, i.e., forest fire model (FFM), Watts-Strogatz (WS) and mixed model under degree, CC, and path length distributions, global CC (GCC), assortativity and modularity and analyzed their performance for various graph types and properties, and concluded that algorithms aggressively exploring the sample node's neighborhood better preserve structural properties and the selection of high-degree nodes is beneficial.

Limitations. Despite several studies, none characterize these sampling algorithms thoroughly under diverse graph properties. We try to fill this gap by analyzing correlations between quality metrics and graphs' size and topological features on six synthetic data types and twelve real graphs.

4 SAMPLING EVALUATION METRICS

We analyze the performance of a sampling algorithm under quality metrics, assessing the similarity of the sample to the original graph under a desired property to preserve.

4.1 Graph Properties

We considered three popular structural graph properties as sampling quality metrics.

(1) *Degree distribution* captures the overall degree structure in the graph in terms of the number of edges connected to each node.
(2) *CC distribution* evaluates the clustering property around every node formulated as the number of closed triangles divided by the possible (closed or open) number of triangles.
(3) *Hop-plot distribution* evaluates the closeness of interconnected nodes (similar to the shortest path) [12, 15] by counting the number of pairs separated by a maximum number of hops.

4.2 Distributions Divergence

Among the different distribution divergence metrics in the literature, we consider the *Kolmogorov-Smirnov D-statistic* metric used in previous studies [12, 29] for analyzing samplings:

$$KS = \left|\max\left(F_G(x) - F_{Gs}(x)\right)\right|,$$

where G and Gs are original and sample graphs and $F_G(.)$ is the cumulative distribution function of graph G. We normalize the distributions to be independent of graph size and capture structural properties, similar to [12]. We analyze sampling algorithms using three quality metrics based on this definition: degree (D3), CC (C2D2), and hop-plots (HPD2) distribution divergences.

5 EXPERIMENTAL DESIGN

We describe the extracted graph features, datasets, and experimental settings in our experiments.

Type	$\vert E\vert$	$\vert G\vert$	$\overline{Deg}$	D	$\overline{CC}$	$\overline{EBC}$	$\frac{\vert N\vert}{\vert E\vert}$	$H(Deg)$	$H(CC)$	$\overline{EIC}$	Dia
AB	$196 \sim 640,000$	460	115	0.13	0.19	2868	0.2	3.60	1.58	0.04	4.70
ER	$1 \sim 800,479$	560	111	0.12	0.12	2298	1.67	2.71	0.65	0.04	5.86
WS	$200 \sim 800,000$	460	135	0.15	0.23	6322	0.20	2.60	1.51	0.04	13.02
PLC	$196 \sim 5991$	480	5	0.02	0.32	3552	0.42	1.84	2.32	0.03	6.75
FFM	$104 \sim 1,801,233$	464	164	0.22	0.45	6013	0.51	2.77	1.99	0.03	14.28
SBM	$316 \sim 404,879$	475	117	0.13	0.29	1726	0.06	3.96	2.43	0.04	3.46

Table 1: Characteristics of synthetic graphs. (|G|: number of graphs).

Dataset	$\vert N\vert$	$\vert$E$\vert$	$\overline{CC}$	*H(deg)*	*H(CC)*	CC_{var}	EIC_{max}	*Dia*
Bio	924	3239	0.88	2.62	2.62	0.122	0.32	10
Email	1005	16,064	0.54	4.32	3.4	0.063	0.17	7
Pow-1138 bus	1138	1458	0.09	1.68	1.09	0.056	0.41	31
Euroroad	1174	1417	0.02	1.39	0.42	0.007	0.22	62
Soc-Wiki vote	889	2914	0.15	2.7	2.41	0.050	0.29	13
Tech-ISP	2113	6632	0.25	2.55	2.32	0.113	0.20	12
Tech-Topology	34,761	107,720	0.29	1.87	1.64	0.167	0.33	10
Tech-Gnutella	62,586	147,892	0.005	2.08	0.18	0.003	0.04	11
Tech-Caida	190,914	607,610	0.16	2.58	2.06	0.072	0.07	26
Cit-Cora	23,166	89,157	0.27	2.92	2.91	0.082	0.14	20
Cit-HepTh	27,769	352,285	0.31	4.14	3.40	0.049	0.26	15
Cit-HepPh	34,546	420,877	0.28	4.14	3.40	0.043	0.11	14

Table 2: Characteristics of real-world graphs.

5.1 Graph features

We considered several graph size and topology features, their statistics (minimum, maximum, median, mean, variance), and the features' calculation time. These features consist of node and edge numbers ($|N|$ and $|E|$), degree (*deg*), *CC*, and *GCC*, degree and CC entropy ($H(.)$), degree assortativity, density (D), node and edge BC (*NBC* and *EBC*), number and sizes of connected components (*ConCS*), eccentricity (*ECC*), eigenvector (*EIC*), PageRank and farness (*FC*) centralities, maximum spanning tree degrees (*DMST*), diameter (*dia*) and shortest path length (*SPL*).

5.2 Datasets

Synthetic graphs. We generated around 2900 graphs of six types, i.e., AB, WS, ER, power-low-cluster (PLC), stochastic block model (SBM), and FFM with $|N|$ of $100 \sim 2000$, summarized in Table 1. These graph types have different properties, i.e., scale-free (AB and PLC), clustering (WS and PLC), community structure (SBM), evolving pattern (FF), and theoretical implications (ER). For further analysis, we extracted 77 graph features (size and topology) and reported average values of the most relevant ones in Table 1.

Real graphs. We considered twelve publicly available[1] [11, 19] real graphs of various sizes of around 1000 to 190.000 nodes and categories, including power, biological, email, infrastructure, social, citation, and technology (Internet service provider (ISP)) graphs. Table 2 represents their characteristics and relevant features.

5.3 Experimental setup

We conducted two sets of sampling experiments in our analysis:

(1) *Small synthetic graphs* finding correlations between sampling algorithms' performance and graphs' features;
(2) *Small and large real-world graphs* investigating algorithms' behavior according to the correlation results.

[1] http://konect.cc/networks/

We considered sampling rates of 0.1 and 0.3, representing the approximate percentage of graph nodes sampled from the graph. We conducted each sampling experiment for five iterations and reported the average results over different sampling rates, graph types, and sampling iterations. We used the *Pearson correlation coefficient* ρ [16] for quantifying the relationship between the graph quality metrics introduced in Section 4.1 and graph features.

6 EVALUATION RESULTS

We provide analysis and evaluations on synthetic and real graphs.

6.1 Synthetic graphs

We summarize the results for the three quality metrics for four graph types and analyze their dependency on graph properties.

6.1.1 Degree distribution divergence. Figure 1(a) compares only four graph types (AB, ER, WS, and SBM) with similar average densities (see Table 1), illustrating the relatively better performance of most algorithms on AB graphs. XS and FF are the best algorithms.

Correlation analysis. Figure 2(a) represents the highly correlated sampling algorithms and graph features (i.e., $|\rho| > 0.5$), including only some statistics of features. The highest correlated features regardless of algorithms are $EIC_{\max}$, $H(deg)$, CC_{var} and EBC_{med}. We also observed a higher correlation of path-related features (FC, SPL, dia and ECC) with traversal-based algorithms, representing better traversing and degree distribution preservation in graphs with higher path lengths. This feature also impacts RE. EBC, CC_{var} also are more relevant to traversal algorithms, with EIC and $H(deg)$ being relevant to most traversal algorithms (indicating their poor degree preservation on graphs with highly randomized degrees, such as SBM graphs (Figure 1(a))). Density is more relevant to FF and RJ. There is also a high relevance of $DMST$ to RNE.

6.1.2 Clustering coefficient distribution divergence. Figure 1(b) represents a better sampling quality in C2D2 than in D3 metric, with better results for WS graphs. These results indicate the better CC preservation of RN and RD for most cases.

Correlation analysis. Figure 2(b) represents the highly correlated graph features with sampling algorithms' C2D2 results. $H(deg)$, $|N|$, $ConCS_{max}$ and NBC are the most relevant feature for most algorithms. $|N|$, $ConCS_{max}$, and PRC are more correlated with node-based algorithms i.e., RN and RNE. $H(deg)$ and $DMST$ are most relevant to RD, MHRW, and FS traversal algorithms that are biased over higher degree nodes. NBC is important for edge-based algorithms (RE, IRE, and RNE) and RDN.

6.1.3 Hop-plot distribution divergence. Figure 1(c) reveals FF as the best algorithm for almost all four graph types. RJ and MHRW have a low HPD2 for some graph types.

Correlation analysis. Figure 2(c) reveals some interesting high HPD2 correlations with path-related features and EBC. Decreasing path-related features results in lower HPD2 for most algorithms, rising from lost connectivity by sampling (except for SB and FS algorithms). We observed the same pattern for EBC and NBC. Whereas D, CC, $H(deg)$, $DMST$, deg and $|E|/|N|$ are negatively correlated with most algorithms, however, they reverse impact FS and SB. This indicates better distance preservation by decreasing distances in dense or highly clustered graphs.

6.2 Real-world graphs

6.2.1 Degree distribution divergence.

Small graphs. Tables 9 and 3 represent that RJ and FF are the best algorithms. Most traversal-based algorithms have D3 under 0.2 (with below 0.1 for FF) for Road and Bus datasets having high path lengths, high $EIC_{\max}$ and EBC_{med}, low $H(deg)$ and density relevant to most algorithms (Figure 2). RJ has a low D3 for the Bio graph with a high CC_{var} and rather high EIC_{max} relevant to RJ. Almost all algorithms have high D3 for the Email dataset, having a low $EIC_{\max}$ and EBC_{med}, and high $H(deg)$ relevant to all samplings.

	FF	XS	RJ	FS	MHRW	RN	RNE	RE	IRE	RDN	SB	RD
Email	0.21	0.44	0.23	0.74	0.32	0.67	0.84	0.52	0.23	0.26	0.65	0.25
Bio	0.18	0.25	0.12	0.38	0.25	0.74	0.83	0.43	0.15	0.20	0.61	0.18
Bus	**0.05**	**0.09**	0.13	0.12	0.13	0.65	0.64	0.46	0.33	0.52	0.26	0.55
Road	**0.06**	0.18	0.19	0.13	0.20	0.78	0.80	0.64	0.54	0.72	0.41	0.68
Wiki	0.15	0.31	0.16	0.37	0.22	0.61	0.71	0.39	0.18	0.17	0.39	0.25
ISP	0.24	0.37	0.17	0.22	0.22	0.55	0.60	0.32	0.18	0.20	0.26	0.26

Table 3: Average D3 for small real-world graphs

Large graphs. Large graphs revealed similar and different patterns. Overall, RJ, IRE and RD perform better than other samplers for large graphs (tables 4 and 9), where RJ is consistent with small-scale results. FS has a very low D3 for Topology network with high EIC_{max} and CC_{var}. FF has a D3 of 0.1 for the HepPh dataset (and 0.12 for Cora) with a lower EIC_{min} (opposite for Topology) and high D3 for Gnutella, with low $EIC_{\max}$ and CC_{var}, consistent with our findings. Therefore, FF is a better choice for citation than technology graphs. RJ and IRE produce good-quality samples for Cora, Caida and HepTh. Cora and HepTh have a very low $EIC_{\min}$ (also Caida) relevant to these algorithms. In addition, Cora and Caida have higher diameters, correlated with them. RDN better estimates the degree property of HepTh with a rather high $EIC_{\max}$.

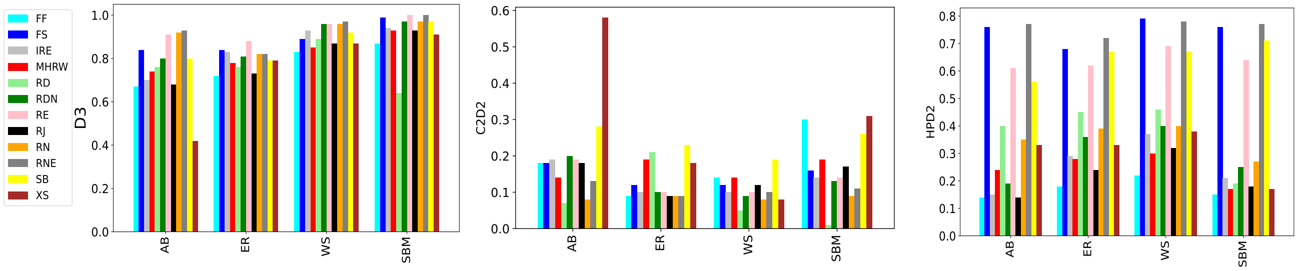
6.2.2 Clustering coefficient distribution divergence.

Small graphs. According to Table 5, RN and RNE have the best results (RN is consistent with synthetic data). Most algorithms can better capture the CC property of Bus, Wiki and ISP networks. It is interpretable for ISP network with more nodes and higher $ConCS_{max}$ relevant to C2D2 of most samplers.

Large graphs. Table 6 illustrates the best results for RN and RNE (as in small-scale). RN has a perfect CC preservation with a maximum C2D2 of 0.01. For Gnutella, most algorithms very well preserve the CC property, having low $H(CC)$ and rather high $|N|$ relevant to C2D2. Additionally, this table represents poor CC preservation of most algorithms on Caida and Topology datasets.

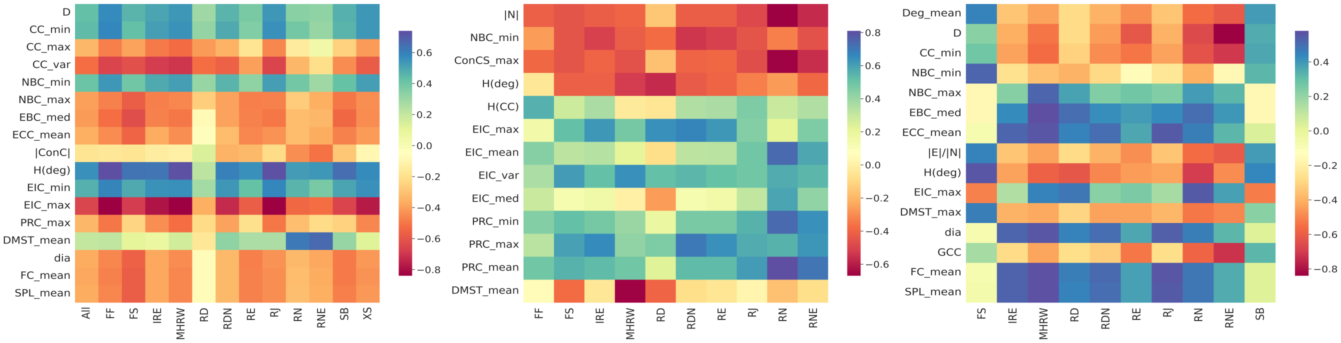
6.2.3 Hop-plot distribution divergence.

Small graphs. Table 7 represents the poor sample quality by almost all algorithms regarding HPD2 on small real graphs, except for SB in Wiki. On average (Table 9), XS, RJ and FF has relatively better results (FF performs well on synthetic (syn) graphs). We also



(a) D3. (b) C2D2. (c) HPD2.

Figure 1: Average synthetic graph quality metric results.



(a) D3. (b) C2D2. (c) HPD2.

Figure 2: Correlation matrix with graph features.

	FF	FS	IRE	MHRW	RD	RDN	RE	RJ	RN	RNE	SB	XS
Gnutella	0.28	0.24	0.27	0.29	0.35	0.24	0.36	0.24	0.56	0.44	0.26	0.33
HepPh	**0.10**	0.77	0.14	0.62	0.17	0.14	0.88	0.16	0.56	0.92	0.63	0.25
HepTh	0.17	0.71	0.12	0.55	**0.08**	0.11	0.83	0.12	0.51	0.89	0.57	0.29
Cora	0.12	0.43	0.11	0.39	**0.10**	0.17	0.69	**0.10**	0.57	0.79	0.41	0.14
Caida	0.24	0.17	**0.10**	0.20	0.14	0.18	0.55	**0.09**	0.55	0.71	0.24	0.20
Topology	0.27	**0.04**	0.25	0.34	0.18	0.28	0.44	0.19	0.65	0.59	0.31	0.29

Table 4: Average D3 for large real-world graphs.

	FF	XS	RJ	FS	MHRW	RN	RNE	RE	IRE	RDN	SB	RD
Email	0.3	0.45	0.26	0.27	0.23	0.11	0.16	0.29	0.28	0.29	0.23	0.34
Bio	0.27	0.36	0.23	0.26	0.31	**0.10**	0.13	0.21	0.19	0.24	0.24	0.34
Bus	0.29	0.13	**0.10**	0.16	0.45	**0.07**	**0.08**	**0.08**	**0.07**	**0.07**	0.26	0.24
Road	0.15	0.29	0.12	0.15	0.38	**0.06**	**0.07**	0.12	0.12	0.11	0.26	0.27
Wiki	0.33	**0.08**	0.15	0.15	0.28	**0.08**	**0.09**	**0.09**	**0.09**	**0.10**	0.12	0.19
ISP	**0.04**	**0.04**	**0.05**	**0.04**	0.12	**0.04**	**0.05**	**0.04**	**0.04**	**0.05**	0.41	0.29

Table 5: Average C2D2 results for small real-world graphs

	FF	FS	IRE	MHRW	RD	RDN	RE	RJ	RN	RNE	SB	XS
Gnutella	0.04	**0.05**	**0.05**	0.19	0.14	**0.05**	**0.05**	**0.04**	**0.01**	**0.02**	**0.1**	**0.07**
HepPh	0.16	0.21	0.21	0.18	0.39	0.22	0.21	0.17	**0.01**	**0.09**	0.31	0.43
HepTh	0.17	0.18	0.17	0.13	0.32	0.18	0.17	0.13	**0.01**	**0.06**	0.33	0.33
Cora	0.18	0.21	0.21	0.28	0.42	0.22	0.21	0.17	**0.01**	**0.09**	0.27	0.37
Caida	0.30	0.28	0.24	0.33	0.47	0.26	0.24	0.18	**0.00**	**0.06**	0.38	0.33
Topology	0.21	0.21	0.23	0.40	0.43	0.3	0.23	0.18	**0.01**	0.12	0.34	0.34

Table 6: Average C2D2 results for large real-world graphs

observed that high path lengths in Road and Bus graphs result in poor HPD2 results for most algorithms.

	FF	XS	RJ	FS	MHRW	RN	RNE	RE	IRE	RDN	SB	RD
Email	0.37	0.55	0.27	0.80	0.15	0.49	0.79	0.49	0.31	0.31	0.53	0.18
Bio	0.31	0.18	0.20	0.46	0.31	0.56	0.87	0.43	0.19	0.26	0.12	0.29
Bus	0.35	0.15	0.56	0.42	0.75	0.96	0.99	0.85	0.71	0.89	0.89	0.92
Road	0.43	0.21	0.63	0.44	0.66	0.97	0.99	0.89	0.80	0.94	0.90	0.94
Wiki	0.30	0.35	0.19	0.42	0.17	0.62	0.91	0.39	0.20	0.27	**0.07**	0.39
ISP	0.36	0.46	0.23	0.52	0.27	0.63	0.95	0.54	0.29	0.37	0.14	0.48

Table 7: Average HPD2 results for small real-world graphs

	FF	FS	IRE	MHRW	RD	RDN	RE	RJ	RN	RNE	SB	XS
HepPh	0.24	0.93	**0.10**	0.63	0.16	**0.10**	0.91	**0.09**	0.53	0.97	0.55	0.28
HepTh	0.29	0.82	0.29	0.52	0.43	0.34	0.83	0.11	0.33	0.94	0.32	0.41
Cora	0.25	0.78	0.11	0.62	0.19	0.11	0.86	0.14	0.47	0.98	0.15	0.11
Topology	0.17	0.33	0.25	0.68	0.28	0.38	0.56	0.20	0.21	0.94	0.33	0.23

Table 8: Average HPD2 results for large real-world graphs.

Large graphs. Table 8 represents HPD2 results for four large real graphs. This table and Table 9 indicate that on average RJ and IRE can better preserve distances (RJ was also good in small graphs). RJ, RDN, and IRE result in low HPD2 for the HepPh with a high $H(deg)$ and low EBC relevant to these algorithms. We observe that most algorithms have lower HPD2 for large graphs. These graphs have lower diameters (Table 2) or path-related features, which is important for most algorithms (Figure 2(c)). Therefore, $H(deg)$, EBC and path-related features appear to be important for the HPD2.

Overall results. The average results of three metrics in Table 9 indicate sampling algorithms' quality regarding scale and type of graphs (for HPD2 metric we only consider four large real graph

	D3			C2D2			HPD2		
	Syn	Real		Syn	Real		Syn	Real	
		Small	Large		Small	Large		Small	Large
FF	0.61	0.15	0.20	0.18	0.23	0.18	0.20	0.35	0.24
FS	0.77	0.33	0.39	0.15	0.17	0.19	0.69	0.51	0.71
IRE	0.71	0.27	0.16	0.15	0.13	0.18	0.28	0.42	0.19
MHRW	0.66	0.22	0.40	0.17	0.30	0.25	0.31	0.38	0.61
RD	0.65	0.38	0.17	0.20	0.42	0.36	0.44	0.56	0.27
RDN	0.79	0.34	0.19	0.16	0.14	0.21	0.34	0.51	0.23
RE	0.89	0.65	0.63	0.15	0.14	0.18	0.66	0.78	0.79
RJ	0.65	0.17	0.15	0.15	0.15	0.15	0.25	0.35	0.13
RN	0.87	0.59	0.57	**0.09**	**0.06**	**0.01**	0.43	0.49	0.38
RNE	0.90	0.74	0.72	0.12	**0.10**	**0.07**	0.78	0.92	0.96
SB	0.79	0.43	0.40	0.23	0.25	0.29	0.60	0.44	0.34
XS	0.60	0.27	0.25	0.36	0.23	0.31	0.30	0.32	0.26

Table 9: Average results for different graph categories

results). The algorithms can better preserve degree distribution for real graphs and many algorithms have better sampling quality for large real graphs. However, regarding CC most algorithms have better sampling quality for synthetic graphs and RN and RNE perform better on large real graphs. Regarding HPD2 most algorithms have better results on large real graphs, due to the lower diameters.

7 CONCLUSION AND FUTURE WORK

We investigated the quality of samples by twelve sampling algorithms of node, edge, and traversal-based categories under D3, C2D2, and HPD2 metrics. We evaluated them using several synthetic graphs of six types and twelve small and large real graphs. Our experiments show different characteristics of algorithms. XS and RJ better capture the degree distribution of synthetic and real graphs respectively. RN results in better samples regarding CC for all graph types. RJ produces better samples regarding hop-plots. Correlation analysis and verification on large real graphs represented the impact of *EIC* (usually high in citation or social networks), path-related features and CC_{var} on D3 results of most algorithms. While, $|N|$ and *ConCS* are relevant to C2D2. $H(deg)$, *EBC* and path-related features are most correlated with HPD2 results. We also discovered inconsistent patterns in large graphs compared with small graphs. As a particular result, the correlation analysis revealed no significant dependency on the sampling rate. Overall, we observed better sample quality of most algorithms on large real graphs under D3 and HPD2 metrics, which is promising for large-scale scenarios.

This work is beneficial to selecting an appropriate sampling algorithm regarding the desired topological property of samples having graph features. It can guide researchers in developing sampling quality predictors by selecting the most relevant features. It can also have implications for understanding algorithms and provide better estimations for original graph properties by considering the most correlated features.

We will conduct more experiments in the future, including larger synthetic and real graphs, other sampling quality metrics, and more sampling algorithms. Furthermore, we will analyze the results using other methods, such as mutual information.

ACKNOWLEDGMENTS

Graph-Massivizer receives funding from the Horizon Europe research and innovation program of the European Union. Its grant management number is 101093202:https://graph-massivizer.eu/.

REFERENCES

[1] Sami Abu-El-Haija et al. 2023. SubMix: learning to mix graph sampling heuristics. In *Uncertainty in Artificial Intelligence*. PMLR.

[2] Nesreen K Ahmed et al. 2013. Network sampling: From static to streaming graphs. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 8 (2013), 1–56.

[3] Simon Geisler et al. 2021. Robustness of graph neural networks at scale. *Advances in Neural Information Processing Systems* 34 (2021), 7637–7649.

[4] Minas Gjoka et al. 2010. Walking in facebook: A case study of unbiased sampling of osns. In *2010 Proceedings IEEE Infocom*. Ieee, 1–9.

[5] Leo A Goodman. 1961. Snowball sampling. *The annals of mathematical statistics* (1961), 148–170.

[6] Pili Hu and Wing Cheong Lau. 2013. A survey and taxonomy of graph sampling. *arXiv preprint arXiv:1308.5865* (2013).

[7] Johannes Illenberger and Gunnar Flötteröd. 2012. Estimating network properties from snowball sampled data. *Social Networks* 34, 4 (2012), 701–711.

[8] Kshitijaa Jaglan et al. 2023. Tight Sampling in Unbounded Networks. *arXiv preprint arXiv:2310.02859* (2023).

[9] Sang Hoon Lee et al. 2006. Statistical properties of sampled networks. *Physical review E* 73, 1 (2006), 016102.

[10] Jure Leskovec et al. 2005. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. 177–187.

[11] Jure Leskovec et al. 2007. Graph evolution: Densification and shrinking diameters. *ACM transactions on Knowledge Discovery from Data (TKDD)* 1, 1 (2007), 2–es.

[12] Jure Leskovec and Christos Faloutsos. 2006. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 631–636.

[13] Arun S Maiya and Tanya Y Berger-Wolf. 2010. Sampling community structure. In *Proceedings of the 19th international conference on World wide web*. 701–710.

[14] Anna Myakushina. [n. d.]. Exploring Sampling Techniques in Large Graphs and Networks. ([n. d.]).

[15] Christopher R Palmer et al. 2002. ANF: A fast and scalable tool for data mining in massive graphs. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. 81–90.

[16] Karl Pearson. 1896. VII. Mathematical contributions to the theory of evolution.—III. Regression, heredity, and panmixia. *Philosophical Transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character* 187 (1896), 253–318.

[17] Bruno Ribeiro and Don Towsley. 2010. Estimating and sampling graphs with multidimensional random walks. In *Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement (IMC '10)*. Association for Computing Machinery, 390–403. https://doi.org/10.1145/1879141.1879192

[18] Bruno Ribeiro and Don Towsley. 2012. On the estimation accuracy of degree distributions from graph sampling. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*. IEEE, 5240–5247.

[19] Ryan A. Rossi and Nesreen K. Ahmed. 2015. The Network Data Repository with Interactive Graph Analytics and Visualization. In *AAAI*. https://networkrepository.com

[20] Benedek Rozemberczki et al. 2020. Karate Club: an API oriented open-source python framework for unsupervised learning on graphs. In *Proceedings of the 29th ACM international conference on information & knowledge management*. 3125–3132.

[21] Daniel A Spielman and Shang-Hua Teng. 2004. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. 81–90.

[22] Michael PH Stumpf et al. 2005. Subnets of scale-free networks are not scale-free: sampling properties of networks. *Proceedings of the National Academy of Sciences* 102, 12 (2005), 4221–4224.

[23] E Upfal and M Mitzenmacher. 2005. Probability and computing.

[24] Elli Voudigari et al. 2016. Rank degree: An efficient algorithm for graph sampling. In *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 120–129.

[25] Yanhong Wu et al. 2016. Evaluation of graph sampling: A visualization perspective. *IEEE transactions on visualization and computer graphics* (2016).

[26] Sooyeon Yoon et al. 2007. Statistical properties of sampled networks by random walks. *Physical Review E* 75, 4 (2007), 046114.

[27] Muhammad Irfan Yousuf et al. 2023. Empirical characterization of graph sampling algorithms. *Social Network Analysis and Mining* 13, 1 (2023), 66.

[28] Muhammad Irfan Yousuf and Suhyun Kim. 2018. List sampling for large graphs. *Intelligent Data Analysis* 22, 2 (2018), 261–295.

[29] Fangyan Zhang et al. 2015. A visual and statistical benchmark for graph sampling methods. In *Exploring Graphs at Scale Workshop*, Vol. 3.