A Sampling-based Framework for Hypothesis Testing on Large Attributed Graphs

Yun Wang
The University of Hong Kong
carrie07@connect.hku.hk

Chrysanthi Kosyfaki
The University of Hong Kong
kosyfaki@cs.hku.hk

Sihem Amer-Yahia
CNRS, Univ. Grenoble Aples
sihem.amer-yahia@univ-grenoble-alpes.fr

Reynold Cheng
The University of Hong Kong
ckcheng@cs.hku.hk

ABSTRACT

Hypothesis testing is a statistical method used to draw conclusions about populations from sample data, typically represented in tables. With the prevalence of graph representations in real-life applications, hypothesis testing on graphs is gaining importance. In this work, we formalize node, edge, and path hypotheses on attributed graphs. We develop a sampling-based hypothesis testing framework, which can accommodate existing hypothesis-agnostic graph sampling methods. To achieve accurate and time-efficient sampling, we then propose a Path-Hypothesis-Aware SamplEr, PHASE, an m-dimensional random walk that accounts for the paths specified in the hypothesis. We further optimize its time efficiency and propose $\text{PHASE}_{\text{opt}}$. Experiments on three real datasets demonstrate the ability of our framework to leverage common graph sampling methods for hypothesis testing, and the superiority of hypothesis-aware sampling methods in terms of accuracy and time efficiency.

PVLDB Reference Format:
Yun Wang, Chrysanthi Kosyfaki, Sihem Amer-Yahia, and Reynold Cheng. A Sampling-based Framework for Hypothesis Testing on Large Attributed Graphs. PVLDB, 17(11): 3192 - 3200, 2024.
doi:10.14778/3681954.3681993

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/Carrieww/GraphHT.

1 INTRODUCTION

Hypothesis testing finds widespread application in various domains such as marketing, healthcare, and social science [23]. Hypotheses are usually tested on representative samples since it is impractical, or even impossible to extract data from entire populations due to size, accessibility or cost. For instance, snowball sampling has been proven effective in accessing a hidden population like drug abusers through a chain-referral mechanism [14, 33]. In this paper, we study hypothesis testing on graphs.

Proceedings of the VLDB Endowment, Vol. 17, No. 11 ISSN 2150-8097.
doi:10.14778/3681954.3681993

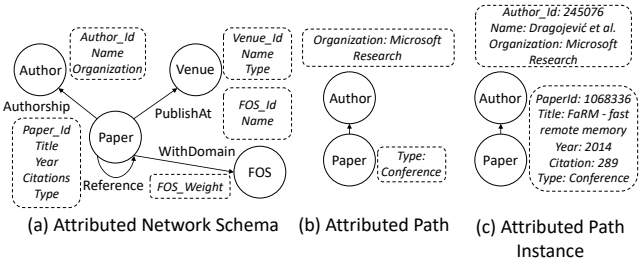


Figure 1: DBLP network schema and paths

Graphs can represent real-world applications such as social, bibliographic, and transportation networks where entities are nodes and edges reflect relationships between them. We focus on attributed graphs that contain multiple types of nodes and edges, each labeled with attributes that provide valuable information for graph analytics [8, 25]. In a bibliographic network like DBLP (see Figure 1a), hypotheses expressed on nodes, edges, and paths unveil interesting aspects of collaboration behavior and research trends based on citation patterns. A node hypothesis examines attributes of a single node type. For example, "the average citation of conference papers is greater than average" concerns the citation attribute of the conference paper nodes. Path hypotheses consider attributes along a defined path, such as a path connecting authors from Microsoft to their conference papers (see Figure 1b and 1c for the path and an path instance, respectively). An example of path hypothesis is "the minimum number of citations of conference papers written by Microsoft researchers is greater than average".

Objectives and Challenges. Given an attributed graph $\mathcal{G}$, our aim is to verify a hypothesis H and return a true or false result. We aim to achieve two objectives: **O1** - ensure hypothesis testing is as accurate as possible; **O2** - minimize execution time. To the best of our knowledge, there is no existing literature addressing hypotheses on attributed graphs. A straightforward approach is to sample a subgraph $\mathcal{S}$ from $\mathcal{G}$ using *hypothesis-agnostic* sampling methods [2, 21, 22]. Sampling is important when the full graph is not accessible or when the hypothesis targets very specific nodes, edges, or paths that are time-consuming to collect. However, hypothesis-agnostic samplers struggle to achieve both objectives. First, hypothesis-agnostic samplers may miss *relevant* nodes, edges, or paths, i.e., those requested by the hypothesis, especially when the sampling budget B, i.e., the size of $\mathcal{S}$, is small. The irrelevant nodes, edges, or paths, which must be filtered out for hypothesis

testing, can compromise accuracy. This raises the accuracy challenge behind **O1**. We show in Section 3.2.3 that the **hypothesis estimator**, which computes aggregate values of nodes, edges, or paths specified in H, *converges to* the ground truth in $\mathcal{G}$ as B increases, resulting in accuracy approaching one. However, the rate of convergence depends on the sampler, presenting a time efficiency challenge behind **O2**. By optimizing the sampler design, we aim to balance accuracy and time efficiency, enabling earlier sampling halts with accurate hypothesis testing results.

Contributions. We classify hypotheses on attributed graphs into three types: node, edge, and path ones. We develop a sampling-based hypothesis testing framework, which accommodates common hypothesis-agnostic samplers, such as random node sampler [29], random walk [13], and non-backtracking random walk [19].

The lack of awareness of the input hypothesis in existing sampling methods may slow down the rate of convergence of accuracy. Therefore, to address objectives **O1** and **O2**, we design PHASE, a Path-Hypothesis-Aware SamplEr. PHASE *is aware of* H and preserves the corresponding nodes, edges, or paths. Consequently, the resulting $\mathcal{S}$ is more likely to accurately test the hypothesis. PHASE employs $m \geq 1$ dependent random walks with two weight functions to ensure path-hypothesis-awareness. One function prioritizes the seed selection toward the first node in the path hypothesis. The other steers the transition probability towards the nodes specified in the hypothesis. We further propose $\text{PHASE}_{\text{opt}}$ to improve the time efficiency of PHASE. $\text{PHASE}_{\text{opt}}$ adopts a non-backtracking approach to avoid selecting previously visited nodes. It also reduces computation overhead by fixing the number of neighbors to examine. We show both theoretically and empirically that, for all samplers, the hypothesis estimator *converges to* the ground truth as B increases, and our proposed samplers ensures earlier and smoother convergence of the hypothesis estimator.

We conduct extensive experiments on three real-world datasets: MovieLens, DBLP, and Yelp. We aim to demonstrate the effectiveness of two optimizations in $\text{PHASE}_{\text{opt}}$ compared to PHASE, and to compare the test significance, accuracy, and execution time of hypothesis-agnostic and hypothesis-aware samplers.

We observe $\text{PHASE}_{\text{opt}}$ is at least 43 times faster than PHASE with less than 4% accuracy difference in DBLP. For significance, $\text{PHASE}_{\text{opt}}$ consistently delivers the most precise and reliable estimates. Compared to 11 state-of-the-art hypothesis-agnostic samplers, we find $\text{PHASE}_{\text{opt}}$ excels in accuracy when B is fixed across various hypothesis types and datasets. It also demonstrates robust accuracy performance, especially for difficult hypotheses, i.e., those with longer paths or fewer relevant nodes, edges, or paths in $\mathcal{G}$. Moreover, the high accuracy achieved in the shortest amount of time makes our proposed sampler usable in practice.

2 DEFINITIONS

2.1 Attributed Graphs

Definition 1 (Attributed Graph). *An attributed graph is a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ denotes the set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges, represented by ordered pairs of nodes. It has a node type mapping function $\phi : \mathcal{V} \to \mathcal{T}$ and an edge type mapping function $\psi : \mathcal{E} \to \mathcal{R}$. Each node and edge has attributes.*

Each node v in $\mathcal{G}$ belongs to a specific node type $\phi(v) \in \mathcal{T}$, and each edge $e = (u, v)$ connecting nodes of type $\phi(u)$ to $\phi(v)$ belongs to a specific edge type $r = \psi(e) \in \mathcal{R}$. Once the relation r exists, its inverse relation r^{-1} naturally holds from $\phi(v)$ to $\phi(u)$. In this work, we assume each node in $\mathcal{G}$ has at least one incoming or outgoing edge. It implies the connectedness of the graph.

The DBLP Network in Figure 1a is an attributed graph containing four types of nodes and four types of edges. For example, paper nodes belong to the node type $\texttt{paper} \in \mathcal{T}$ with five attributes, including title and year. Edges from paper to field of study (FOS) nodes belong to the edge type $\texttt{WithDomain} \in \mathcal{R}$ with a weight attribute indicating the paper's relevance to an FOS. The reverse relationship $\texttt{WithDomain}^{-1} \in \mathcal{R}$ holds accordingly from FOS to paper nodes.

Definition 2 (Path). *A path $\mathcal{P}$ is defined as*

$$t_1 \xrightarrow{r_1} \ldots \xrightarrow{r_l} t_{l+1}$$

where the node type t_i and edge type r_i can repeat; $l \geq 0$ is the length of $\mathcal{P}$. When $l = 0$, $\mathcal{P}$ is a node and when $l = 1$, it is an edge.

An **attributed path** is a path where each node has some attributes, referred to as **modifiers**. Figure 1b presents an example of a length-one attributed path, "conference papers written by Microsoft researchers", and Figure 1c is an instance of that path.

2.2 Hypotheses on Attributed Graphs

We formally define path hypotheses on attributed graphs. Node and edge hypotheses are two special cases of path hypotheses.

Definition 3 (Path Hypothesis). *Given a path $\mathcal{P} = t_1 \xrightarrow{r_1} t_2 \xrightarrow{r_2} \ldots \xrightarrow{r_l} t_{l+1}$ in $\mathcal{G}$, where $l \geq 0$, a path hypothesis is defined as:*

$$H_{path} : P_c^o \left(agg(f_{\mathcal{P}} \mid M_{t_i}, \forall t_i \text{ on } \mathcal{P}) \right)$$

where $o \in \{=, <>, >, <\}$; $c \in \mathbb{R}$ is a constant value; P_c^o is the predicate in the format: equal, unequal, greater, or less than a value; $f_{\mathcal{P}}$ is any function of node and/or edge attributes on $\mathcal{P}$; $agg(f_{\mathcal{P}} \mid M_{t_i}, \forall t_i \text{ on } \mathcal{P})$ is an aggregation function applying on the $f_{\mathcal{P}}$ of all paths whose nodes satisfy the corresponding modifiers.

In the DBLP network defined previously, co-authorship can be represented as a path:

$$\mathcal{P} = \texttt{author} \xrightarrow{\texttt{Authorship}^{-1}} \texttt{paper} \xrightarrow{\texttt{Authorship}} \texttt{author}$$

"The average citation of papers co-authored by Microsoft researchers is greater than 100" can be expressed as:

$$P_{100}^{>} \left(avg(f_{\mathcal{P}} \mid \texttt{author[MSR]}, \texttt{paper[]}, \texttt{author[MSR]}) \right)$$

where $f_{\mathcal{P}} = \texttt{paper[citation]}$, MSR stands for microsoft research.

When $l = 0$, a path hypothesis is reduced to a **node hypothesis**. An example in DBLP, "the average number of citations for conference papers is larger than 50", can be expressed as:

$$P_{50}^{>} \left(avg(\texttt{paper[citation]} \mid \texttt{paper[conference]}) \right)$$

When $l = 1$, we refer it as an **edge hypothesis**. For instance, the edge hypothesis, "the average FOS_weight of conference papers on data mining is larger than 0.5", can be expressed as:

$$P_{0.5}^{>} \left(avg(\texttt{WithDomain[FOS\_Weight]} \mid \texttt{paper[C]}, \texttt{FOS[DM]}) \right)$$

C (resp. DM) stands for conference papers (resp. DM data mining).

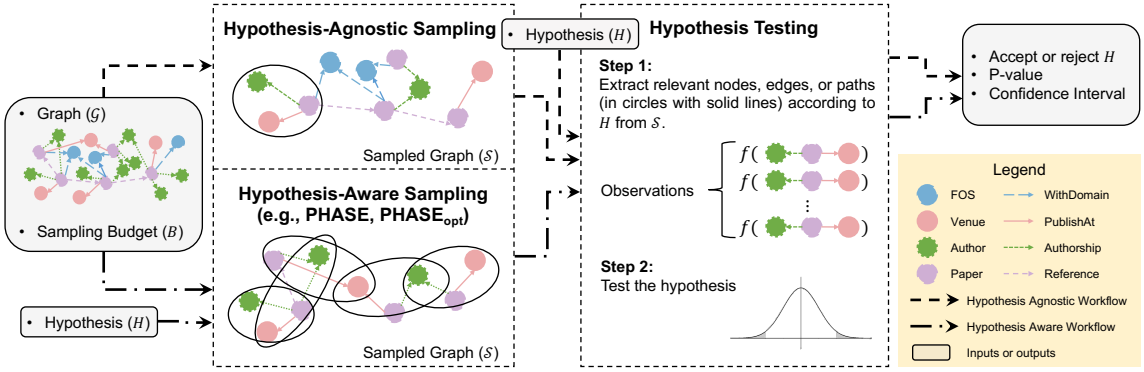


Figure 2: The Sampling-based Hypothesis Testing Framework on attributed graphs.

2.3 Problem Statement and Challenges

Given an attributed graph $\mathcal{G}$, our aim is to verify a node, edge, or path hypothesis H and return true or false. We formulate two objectives to address our problem: **O1** - ensure hypothesis testing is as accurate as possible; **O2** - minimize execution time.

When testing a hypothesis like "the average number of citations for conference papers is greater than 50", a conventional approach is to collect representative conference papers from the database, compute the average number of citations, and perform the statistical test. However, the entire graph may not be accessible, or collecting all relevant nodes, edges, or paths from $\mathcal{G}$ is time-consuming and impractical. Hence, we adopt graph sampling techniques to sample a subgraph $\mathcal{S}$ from $\mathcal{G}$, assuming a sampling budget B that reflects the maximum size of $\mathcal{S}$. For simplicity, unless stated otherwise, sampling an edge or a node incurs the same unitary cost of one.

A hypothesis estimator computes an aggregate value for nodes, edges, or paths based on H. It directly impacts the accuracy of hypothesis testing. The challenge for hypothesis-agnostic samplers to achieve **O1** is that they can miss relevant nodes, edges, or paths when B is small. By examining the applicability of existing hypothesis-agnostic samplers, we observe that a large B (almost the size of $\mathcal{G}$) is required to achieve an accuracy of one, which results in longer execution time and the challenge behind **O2**. We conjecture that this is due to a lack of awareness of the underlying hypothesis during sampling. Hence, to address two challenges, we aim to design a *hypothesis-aware* sampler that preserves relevant nodes, edges, or paths in S. We also intend to show, both theoretically and empirically, that using our sampler, the hypothesis estimator on $\mathcal{S}$ will *converge earlier* to its corresponding value in $\mathcal{G}$ as B increases.

3 SAMPLING-BASED HYPOTHESIS TESTING

We propose a sampling-based hypothesis testing framework that supports both hypothesis-agnostic and hypothesis-aware graph samplers. After reviewing existing hypothesis-agnostic methods, we introduce our hypothesis-aware sampler, PHASE, in Section 3.2.1. It achieves **O1** by incorporating hypothesis awareness into the sampling, and **O2** by ensuring earlier convergence of hypothesis estimators. Additionally, we implement two optimizations in PHASE$_{\text{opt}}$ to reduce the execution time of PHASE in Section 3.2.2.

3.1 Hypothesis-Agnostic Samplers

Figure 2 illustrates our sampling-based hypothesis testing framework for testing node, edge, and path hypotheses on attributed graphs. It consists of two steps: (1) Sampling and (2) Hypothesis Testing. There are two workflows: hypothesis-agnostic, shown with dashed arrows, where H is considered only in the hypothesis testing step, and hypothesis-aware, indicated by dash-dot-dash arrows, which requires H during the sampling step.

In the hypothesis-agnostic workflow (dashed arrows), given $\mathcal{G}$ and B, a hypothesis-agnostic sampler is used in the sampling step to obtain $\mathcal{S}$. Existing hypothesis-agnostic samplers fall into three categories: node samplers that choose B nodes from $\mathcal{G}$ [2, 29], edge samplers that choose B edges from $\mathcal{G}$ [18], and random walk based samplers that pick edges and nodes by random walks [13, 19–21]. In the hypothesis testing step, relevant nodes, edges, or paths are extracted from $\mathcal{S}$ for hypothesis testing. Finally, the acceptance result, p-value, and confidence interval are returned.

3.2 Hypothesis-Aware Samplers

3.2.1 PHASE Algorithm. We introduce PHASE, our Path-Hypothesis-Aware SamplEr (Algorithm 1), for the sampling step. Since $\mathcal{G}$ may contain both relevant and irrelevant nodes, edges, or paths for a given H, PHASE aims to preferentially sample those specified in H. This strategy resembles stratified sampling [7, 24], where the target population is sampled at higher rates without bias. PHASE can be integrated with any random walk based sampler. To clarify, we describe it using Frontier Sampler (*FrontierS*) [28], which picks a node from m random seeds based on degree-proportional probability and performs a random walk by uniformly selecting a neighboring node. This process repeats until B is reached.

PHASE takes $\mathcal{G}$, B, H, Q as inputs. For a node, edge, or path hypothesis, H contains $\mathcal{P}$ with lengths zero, one, or more than one, respectively. m seed nodes in $\mathcal{G}$ are randomly initialized and stored in a list L (Line 1). This line increases the chance of picking relevant nodes, edges, and paths and ensures they are not locally clustered, preventing locality bias. Each seed is assigned a weight in L_w (Line 2) to guide the selection of seed nodes for random walks later. The weights are determined based on heuristics: nodes satisfying the first node modifier on $\mathcal{P}$ (denoted by x_1) receive a higher weight w_h, while others receive a lower weight w_l, where $w_h \geq w_l > 0$.

Algorithm 1 PHASE

Input: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, B, H, Q
Output: a sampled graph $\mathcal{S}$

1: Initialize $L = (v_1, v_2, \ldots, v_m)$ with m randomly chosen nodes
2: L_w = AssignWeight(L, H) ▷ Assign w_h to x_1 nodes and w_l to others.
3: $\mathcal{V}_S = \{\}$, $\mathcal{E}_S = \{\}$
4: **while** $B > m$ **do**
5: Normalize L_w
6: Select $v \in L$ with probability L_w
7: $N \leftarrow N[v]$ ▷ $N[v]$ is the set of neighbors of v.
8: N_w = AssignWeight(N, H, Q) ▷ Assign weights according to Q.
9: Select $u \in N$ with the normalized N_w
10: $\mathcal{V}_S$.update(v, u)
11: Replace v by u in L and update L_w
12: $B = B - 1$
13: **end while**
14: **return** $\mathcal{S} = \{\mathcal{V}_S, \mathcal{E}_S\}$

	x_1	y
x_1	w_h	w_l
y	w_h	w_l

(a)

	x_1	x_2	y
x_1	w_l	w_h	w_l
x_2	w_h	w_l	w_l
y	w_h	w_l	w_l

(b)

	x_1	x_2	x_3	y
x_1, x_1	w_l	w_h	w_l	w_l
x_1, x_2	w_l	w_l	w_h	w_l
x_1, x_3	w_h	w_l	w_l	w_l
x_1, y	w_h	w_l	w_l	w_l

(c)

Figure 3: Transition probability matrices Q for (a) node, (b) edge, and (c) path ($l = 2$) hypotheses (up to the first four rows). x_i represents nodes in $\mathcal{G}$ satisfying the i-th node modifier on $\mathcal{P}$ and y represents other nodes.

Lines 4-14 describe an iterative random walk to select nodes until B is reached. During each iteration, the algorithm chooses a node v from L based on the probability distribution L_w (Line 6). Next, it picks a neighbor u of v using a weighted selection process (Lines 7-9) (**O1**). Specifically, in Line 8, we steer the random walks towards relevant nodes, edges, or paths. The transition probability matrices for node, edge, and path hypotheses are shown in Figures 3a, 3b, and 3c, respectively. For a node hypothesis, a higher weight w_h is assigned to x_1 regardless of the current node. For an edge hypothesis, w_h is given to x_2 when the current node is x_1, and given to x_1 in other cases. This increases the chance of sampling more relevant edges (x_1, x_2). The random walks are 1st-order in Figures 3a and 3b. Path hypotheses ($l = 2$) have 2nd-order random walks (Figure 3c), meaning the transition depends on the current and previous nodes. For example, given two nodes x_1 and x_2, x_3 gets a higher weight w_h. The consistent w_h prevents sampling bias within the population. After adding nodes u and v to $\mathcal{V}_S$ (Line 10), the algorithm updates L by replacing v with u (Lines 11) and decreases the sampling budget by one (Lines 12). This iteration continues until $B > m$. The resulting sampled graph $\mathcal{S}$ is the induced subgraph from $\mathcal{V}_S$.

3.2.2 PHASE$_{opt}$ Algorithm. We further propose optimizations to reduce execution time based on heuristics. Algorithm 2 introduces two lines that replace line 7 in Algorithm 1. In line 1, we employ a non-backtracking approach to avoid selecting previously visited nodes during sampling [19], preventing cycles and unnecessary revisits. Additionally, in densely connected graphs, computing neighbor weights in line 8 of Algorithm 1 can be time-consuming due to the potentially large number of neighbors. To address this, line 2 in Algorithm 2 introduces random sampling of a subset of $\min\{|N'|, n\}$ neighbors as candidates, where n is a parameter. This can effectively reduce computation time. But it may result in some accuracy loss when n is significantly smaller than $|N'|$.

Algorithm 2 PHASE$_{\text{opt}}$
(showing only 2 lines that replace line 7 in Algorithm 1

1: $N' \leftarrow N[v] - \mathcal{V}_S$ (Optim 1)
2: $N \leftarrow$ Select $\min\{|N'|, n\}$ nodes randomly from N' (Optim 2)

Complexity. On average, each node has $2|\mathcal{E}|/|\mathcal{V}|$ neighbors to be examined by PHASE, resulting in a time complexity of $O(B \times 2|\mathcal{E}|/|\mathcal{V}|)$. On the other hand, by constraining the number of neighbors, PHASE$_{\text{opt}}$ achieves a time complexity of $O(B)$.

3.2.3 Convergence of Hypothesis Estimators. We demonstrate the convergence of hypothesis estimators, which ensures the convergence of hypothesis testing accuracy to one, for all sampling methods. Then, we show our proposed sampler achieves earlier and smoother convergence (**O2**).

We will focus on scenarios where agg is an average function to construct and prove the convergence of hypothesis estimators. Estimators for other aggregation functions, such as maximum and minimum, can be derived analogously.

For a path hypothesis ($l \geq 0$), $\text{avg}(f_{\mathcal{P}} \mid M_{t_i}, \forall t_i \text{ on } \mathcal{P})$ has a primary subject $f_{\mathcal{P}}$. Let $\mathcal{P}^*$ be all relevant paths in $\mathcal{G}$, and $\mathcal{P}_{\mathcal{G}}$ be all paths with the same length as $\mathcal{P}$ in $\mathcal{G}$. The mean value of the path hypothesis, θ_{path}, is

$$\theta_{path} = \frac{1}{|\mathcal{P}^*|} \sum_{\forall \mathcal{P} \in \mathcal{P}_{\mathcal{G}}} T(\mathcal{P}) \tag{1}$$

where $T(\mathcal{P}) = f_{\mathcal{P}} \times \mathbb{1}_{M_{t_i} \subseteq \mathcal{L}_{\phi(t_i)} \forall t_i \text{ on } \mathcal{P}}$. Replacing $\mathcal{G}$ with the sampled graph $\mathcal{S}$, the estimator for θ_{path} on $\mathcal{S}$ is

$$\hat{\theta}_{path} = \frac{1}{|\mathcal{P}^* \cap \mathcal{P}_{\mathcal{S}}|} \sum_{\forall \mathcal{P} \in \mathcal{P}_{\mathcal{S}}} T(\mathcal{P}) \tag{2}$$

When $l = 0$ (resp. $l = 1$), we name the corresponding mean value θ_{node} (resp. θ_{edge}) and estimator $\hat{\theta}_{node}$ (resp. $\hat{\theta}_{edge}$).

Some sampling methods, including random edge sampler, simple random walk, and *FrontierS*, obey the Strong Law of Large Numbers (SLLN) [28]. Their $\hat{\theta}_{node}$ and $\hat{\theta}_{edge}$ are asymptotically unbiased estimators of θ_{node} and θ_{edge}, respectively, when $B \to \infty$.

Theorem 1 (SLLN). *For any function f, where $\sum_{(u,v)\in\mathcal{E}} |f(u,v)| < \infty$,*

$$\lim_{B\to\infty} \frac{1}{B} \sum_{i=1}^{B} f(u_i, v_i) \to \frac{1}{|E|} \sum_{\forall (u,v)\in\mathcal{E}} f(u,v)$$

almost surely, i.e. the event occurs with probability one.

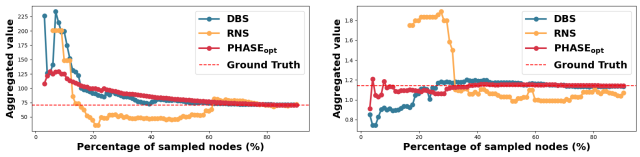


Figure 4: The convergence of hypothesis estimator for two path hypotheses: DB-P3 (left) and YP-P3 (right).

In general scenarios with any path hypotheses ($l \geq 0$) and sampler, SLLN may not apply, and the estimator may not be asymptotically unbiased. However, $\hat{\theta}_{\text{path}}$ still converges to θ_{path} due to Finite Population Correction (FPC), a statistical adjustment made when sampling without replacement from a finite population [6]. As B increases, FPC $= \frac{|\mathcal{P}^*|-\mathcal{P}_S}{|\mathcal{P}^*|-1}$ approaches zero, which ensures the convergence of the hypothesis estimator to its ground truth.

While all samplers ensure the convergence of the hypothesis estimator, our proposed samplers achieve earlier convergence than other methods. This is due to their higher probability of selecting relevant nodes, edges, and paths. For instance, when $l = 2$, the probability to pick nodes (x_1, x_2, x_3) is:

$$P(x_1, x_2, x_3) = P(x_1) \times (P(x_2 \mid x_1) \times A(x_2 \mid x_1)) \times (P(x_3 \mid x_1, x_2) \times A(x_3 \mid x_1, x_2))$$

where $A(x_i \mid x_{i-1})$ indicates the accessibility (0 or 1) of node x_i from x_{i-1}. Assuming node x_i has n_i neighbors with k_i of type x_{i+1} and there are d x_1 in m seed nodes, where $d \geq 0$ and $k_i \geq 0$:

$$P_{\text{PHASE}}(x_1, x_2, x_3) = \frac{d \cdot w_h}{d \cdot w_h + (m-d) \cdot w_l} \times \left(\frac{w_h}{k_1 \cdot w_h + (n_1 - k_1) \cdot w_l} \times A(x_2 \mid x_1) \right) \times \left(\frac{w_h}{k_2 \cdot w_h + (n_2 - k_2) \cdot w_l} \times A(x_3 \mid x_1, x_2) \right)$$

Random node and edge samplers have the lowest probability of picking (x_1, x_2, x_3). Random walk based samplers have a transitional probability from x_i of $\frac{1}{n_i} \leq \frac{w_h}{k_i \cdot w_h + (n_i - k_i) \cdot w_l}$, particularly when k_i is large. With a higher probability to pick (x_1, x_2, x_3), PHASE and $\text{PHASE}_{\text{opt}}$ have earlier convergence of the hypothesis estimator than existing hypothesis-agnostic samplers. Also, as shown in Figure 4, the empirical results of two path hypotheses, whose details are in Table 2, align with the claim.

4 EXPERIMENTS

The goal of our experiments is twofold: 1) test the effectiveness of optimizations in $\text{PHASE}_{\text{opt}}$ compared to PHASE (Section 4.3), and 2) compare hypothesis-agnostic and hypothesis-aware samplers in terms of test significance (Section 4.4), accuracy (Section 4.5), and time (Section 4.6).

4.1 Experimental Setup

Datasets. We use three datasets [1, 15, 31], extracted from real attributed networks: MovieLens, DBLP, and Yelp. Table 1 shows the statistics of the datasets.

Table 1: Statistics of datasets

Dataset	#(Nodes)	#(Edges)	Density	#(Node Types)	#(Edge Types)
MovieLens	9,705	996,656	1.06e-02	2	1
DBLP	1,623,013	11,040,170	4.19e-06	4	4
Yelp	2,136,118	6,743,879	1.48e-06	2	1

Samplers We compare $\text{PHASE}_{\text{opt}}$ with 11 existing samplers:

- Node samplers: Random Node Sampler (*RNS*) [29], Degree-Based Sampler (*DBS*) [2]
- Edge samplers: Random Edge Sampler (*RES*) [18]
- Random walk based samplers: Simple Random Walk (*SRW*) [13], Frontier Sampler (*FrontierS*) [28], Non-Backtracking Random Walk (*NBRW*) [19], Random Walk with Restarter (*RWR*) [20], Metropolis-Hastings Random Walk (*MHRW*) [30], Snaw Ball Sampler (*SBS*) [14], Forest Fire Sampler (*FFS*) [20], Shortest Path Sampler (*ShortestPathS*) [27]

Hypotheses. For each dataset (e.g., DB) and hypothesis type (e.g., N for node hypotheses), we chose three example hypotheses (e.g., DB-N1) in the experiment. Due to the page limit, examples for DBLP and Yelp are shown in Table 2. These hypotheses are chosen based on context and difficulty. Based on the context of the datasets, the subject of interest and the constant in the hypothesis should reflect meaningful requests from real users. The difficulty is related to the path length and the number of relevant nodes, edges, or paths, as indicated in the third column of Table 2. The longer the path or the fewer the relevant nodes, edges, or paths, the more difficult it becomes to sample them from $\mathcal{G}$ for accurate hypothesis testing. We use the following path hypotheses of DBLP with lengths of three and four in the experiment: "the average citation of two papers, each authored by a Microsoft researcher and citing each other, > 50" and "the average citation of two conference papers, each authored by a Microsoft researcher, > 50".

Parameter Choice. We determine optimal settings for m, n, w_h, and w_l through a grid search, with m and n ranging from 10 to 200, and w_h and w_l from 0.1 to 20. We set $m = 50$ to maintain the path-preserving ability of multi-dimensional random walks, and $n = 30$ to strike a balance between accuracy and time efficiency. $w_h = 10$ and $w_l = 0.1$ help regulate the prioritization towards sampling relevant nodes, edges, or paths. The sampling budget B is maintained as a proportion of the total number of nodes in $\mathcal{G}$ for all samplers. In real deployment, we recommend iteratively increasing B until the accuracy stabilizes at a high threshold (e.g., 0.9) based on the average from 30 samples to determine the optimal B. As for existing hypothesis-agnostic samplers, we use the best parameters in their respective settings. We report an average of 30 runs for every evaluation measure.

4.2 Evaluation Measures

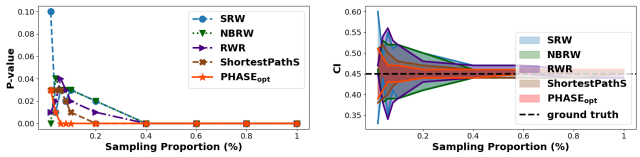
Accuracy. Accuracy at a specific sampling proportion reflects the effectiveness of a sampling method. It measures the number of matched hypothesis testing results on $\mathcal{G}$ and $\mathcal{S}$:

$$\text{Accuracy} = \frac{1}{k} \sum_{k} \mathbb{1}_{H(\mathcal{G})==H(\mathcal{S})}$$

where $H(\mathcal{G})$ and $H(\mathcal{S})$ return 0 (false) or 1 (true).

Table 2: Examples of hypotheses for DBLP and Yelp

Hypothesis Type	Example	Relevant nodes, edges, paths
DBLP		
Node	**DB-N1:** The avg citation of papers published as journals > 20	199205 (easy)
	DB-N2: The avg citation of papers in conferences in 2010 > 10	31566 (medium)
	DB-N3: The avg citation of papers published in Journal in 2017 > 10	1588 (hard)
Edge	**DB-E1:** The avg weight of conference papers on data mining > 0.5	44925 (easy)
	DB-E2: The avg weight of journal papers on data mining > 0.5	13400 (medium)
	DB-E3: The avg weight of papers on telecommunications network > 0.5	2510 (hard)
Path	**DB-P1:** The avg weight of papers by China's institutes on data mining > 0.5	17671 (easy)
	DB-P2: The avg citation of papers co-authored by authors in Peking and China's institutions > 50	7065 (medium)
	DB-P3: The avg citation of conference papers by Microsoft Researchers > 50	3217 (hard)
Yelp		
Node	**YP-N1:** The avg reviews given by users with high popularity > 200	112043 (easy)
	YP-N2: The avg stars given by users who have low prolificacy and medium popularity > 3	16429 (medium)
	YP-N3: The avg number of reviews of business in Illinois > 4	2144 (hard)
Edge	**YP-E1:** The avg ratings of fast food > 4	224536 (easy)
	YP-E2: The avg ratings of furniture stores > 3	33040 (medium)
	YP-E3: The avg percentage of useful reviews given by useful writers to Illinois businesses > 0.5	4242 (hard)
Path	**YP-P1:** The avg rating difference on path [business in FL - high popularity user - business in LA] > 0.5	615174 (easy)
	YP-P2: The avg rating difference on path [business in LA - high popularity user - business in IL] > 0.5	15542 (medium)
	YP-P3: The avg rating difference on path [business in LA - medium popularity user - business in AB] > 0.5	1080 (hard)



(a) P-value Plot **(b) CI Plot**

Figure 5: DBLP p-value and CI plot for the hypothesis DB-P1

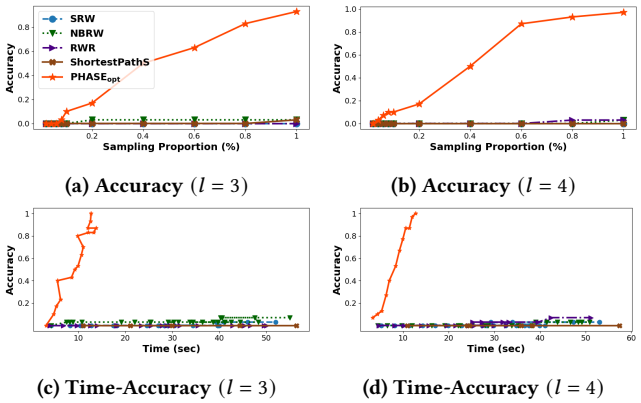
Time. We measure the total execution time, including the sampling time, the time taken to extract relevant information from $\mathcal{S}$ for hypothesis testing, and hypothesis testing time.

4.3 PHASE vs PHASE$_{opt}$

We evaluate PHASE$_{opt}$'s optimizations compared to PHASE in Table 3 and 4. For DBLP, PHASE$_{opt}$ is at least 43 times faster than PHASE with minimal accuracy loss (under 4%). Consequently, we use PHASE$_{opt}$ exclusively in further experiments and discussions.

4.4 Significance

To assess statistical significance and estimation precision, we evaluate p-value trends and confidence intervals (CIs) as B increases using DB-P3, as shown in Figure 5. Similar trends are observed for



(a) Accuracy ($l = 3$) **(b) Accuracy** ($l = 4$)

(c) Time-Accuracy ($l = 3$) **(d) Time-Accuracy** ($l = 4$)

Figure 6: Accuracy (a, b) and time-accuracy (c, d) performance for DBLP path hypotheses.

other hypotheses. PHASE$_{opt}$ consistently maintains p-values below the significance level (e.g., 0.05), showing stronger evidence against the null hypothesis as B increases. It also exhibits the narrowest CI at all B, reflecting the highest precision among other samplers.

4.5 Accuracy

Table 3 presents the accuracy of 11 existing samplers and PHASE$_{opt}$ with a fix B in column three. B is set to ensure the accuracy is sufficiently stabilized without saturating across all samplers. Each row reports an average accuracy from three examples, based on 30 runs each. The highest and second highest accuracies are highlighted in bold and underlined, respectively.

Table 3 shows PHASE$_{opt}$'s robust performance across most scenarios, except for MovieLens edge hypotheses where it slightly lags behind *NBRW*. *SRW*, *RWR*, and *ShortestPathS* perform well on edge and path hypotheses, whereas *RES*, *RNS*, and *DBS* struggle with path hypotheses. This observation aligns with their sampling mechanisms: node and edge samplers can hardly preserve the path information from $\mathcal{G}$. Also, PHASE$_{opt}$ outperforms *FrontierS*, showcasing the effectiveness of the two weight functions in enhancing the sampler's hypothesis-awareness.

We rank the accuracy of all samplers by averaging their accuracy per column in Table 3 and identify the top five: PHASE$_{opt}$, *NBRW*, *ShortestPathS*, *RWR*, and *SRW*. The accuracy performance on individual hypotheses for DBLP and Yelp is shown in Figures 7 and 8, respectively. Figures for MovieLens are omitted due to the space constraints and result similarity. Subfigures a-c, e-g, and i-k depict accuracy versus B for three node, edge, and path hypotheses, respectively, with truncated x-axes to highlight the convergence.

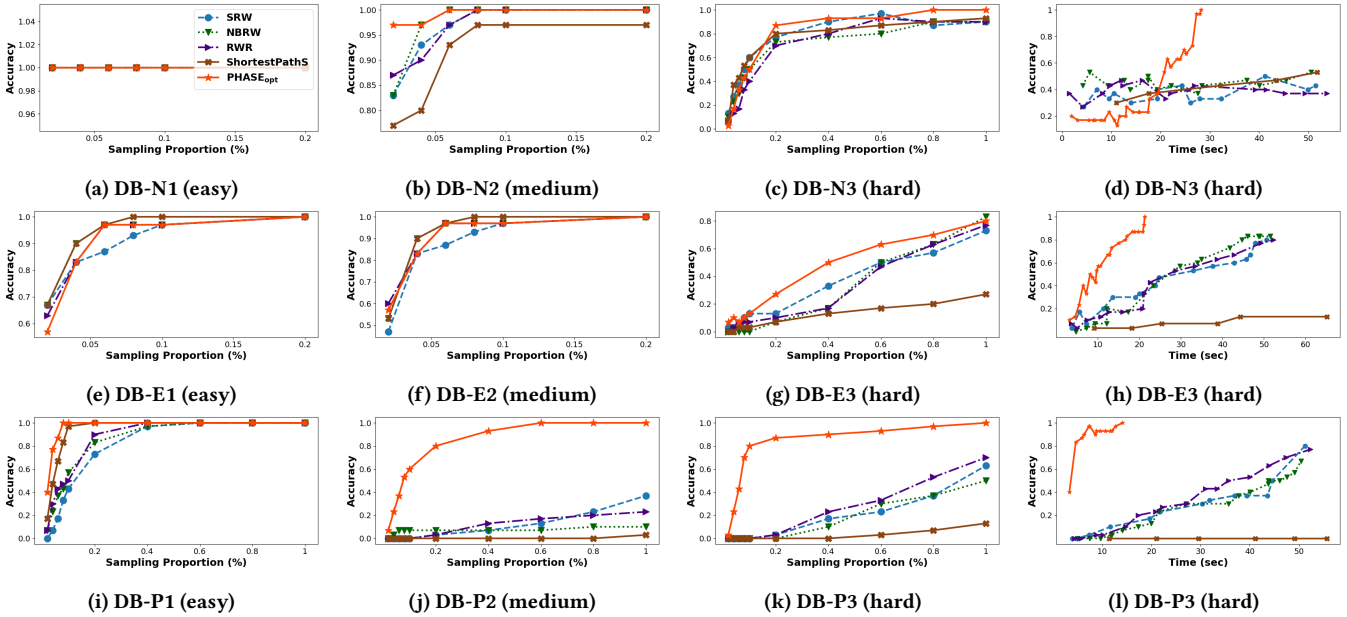
In Figures 7 and 8, PHASE$_{opt}$ consistently outperforms other samplers across most hypotheses and sampling proportions. When there are abundant relevant nodes, edges, or paths in $\mathcal{G}$ (e.g, easy and medium hypotheses), hypothesis-agnostic samplers can perform well, reducing PHASE$_{opt}$'s relative advantage. However, its superiority is more obvious for more difficult hypotheses (e.g., hard hypotheses). Also, for difficult hypotheses with longer paths in DBLP, PHASE$_{opt}$ achieves the highest accuracy at any sampling proportion, as shown in Figures 6a and 6b. Moreover, *ShortestPathS*

Table 3: The accuracy of 11 existing samplers and PHASE$_{opt}$ on three datasets and three types of hypothesis.

Dataset	Hypothesis Type	Sampling Proportion (%)	PHASE$_{opt}$	PHASE	RES	RNS	DBS	SRW	NBRW	RWR	MHRW	ShortestPathS	FrontierS	FFS	SBS
MovieLens	Node	1	0.9	**1**	0.89	0.83	0.86	0.88	0.89	0.87	0.89	0.86	0.84	0.83	0.56
	Edge	2.5	0.98	**1**	0.68	0.77	0.98	0.99	**1**	0.99	0.98	0.99	0.73	0.93	0.91
	Path	5	0.99	**1**	0.1	0.88	0.83	0.82	0.89	0.91	0.98	0.95	0.38	0.85	0.62
DBLP	Node	0.2	0.96	**1**	0.48	0.87	0.93	0.92	0.91	0.9	0.94	0.92	0.92	0.94	0.88
	Edge	0.2	0.76	**0.79**	0.48	0	0.71	0.73	0.69	0.7	0.29	0.69	0.63	0.7	0.42
	Path	0.2	**0.89**	0.88	0	0	0	0.26	0.3	0.32	0.18	0.33	0.043	0.3	0.12
Yelp	Node	0.1	0.99	**1**	0.65	0.77	0.61	0.69	0.69	0.69	0.77	0.7	0.64	0.66	0.48
	Edge	1	**1**	**1**	0.73	0.54	0.76	0.79	0.91	0.87	0.84	0.76	0.77	0.79	0.71
	Path	1	0.99	**1**	0.11	0.05	0.79	0.78	0.78	0.72	0.8	0.99	0.42	0.67	0.42

Table 4: The execution time (sec) of 11 existing samplers and PHASE$_{opt}$ on three datasets and three types of hypothesis.

Dataset	Hypothesis Type	Sampling Proportion (%)	PHASE$_{opt}$	PHASE	RES	RNS	DBS	SRW	NBRW	RWR	MHRWS	ShortestPathS	FrontierS	FFS	SBS
MovieLens	Node	1	0.083	0.41	0.99	**0.023**	0.077	0.057	0.06	0.05	0.06	0.063	0.083	0.067	0.047
	Edge	2.5	0.45	2.07	0.99	**0.11**	0.36	0.31	0.36	0.33	0.29	0.26	0.33	0.35	0.31
	Path	5	4.92	14.80	1.03	**0.34**	4.32	4.53	4.55	4.76	3.10	0.95	0.38	3.87	3.08
DBLP	Node	0.2	5.56	418.07	18.70	**0.55**	6.98	7.98	8.22	9.67	1.66	31.32	10.48	5.73	3.03
	Edge	0.2	8.76	414.47	22.57	**0.90**	10.07	14.76	12.89	12.32	3.53	33.91	14.46	8.87	5.27
	Path	0.2	5.44	236.82	19.61	**0.71**	6.87	8.24	8.39	8.13	2.17	31.13	9.37	5.33	3.01
Yelp	Node	0.1	2.56	6.10	13.84	1.02	6.81	1.19	**0.96**	1.16	1.12	1.84	1.68	1.02	6.58
	Edge	1	19.42	67.55	16.88	**2.01**	13.61	8.58	8.97	9.17	6.93	30.23	10.22	10.27	6.03
	Path	1	56.97	130.3	14.35	**1.53**	23.48	15.97	16.61	19.03	7.35	37.81	9.95	25.61	19.94



(a) DB-N1 (easy) (b) DB-N2 (medium) (c) DB-N3 (hard) (d) DB-N3 (hard)

(e) DB-E1 (easy) (f) DB-E2 (medium) (g) DB-E3 (hard) (h) DB-E3 (hard)

(i) DB-P1 (easy) (j) DB-P2 (medium) (k) DB-P3 (hard) (l) DB-P3 (hard)

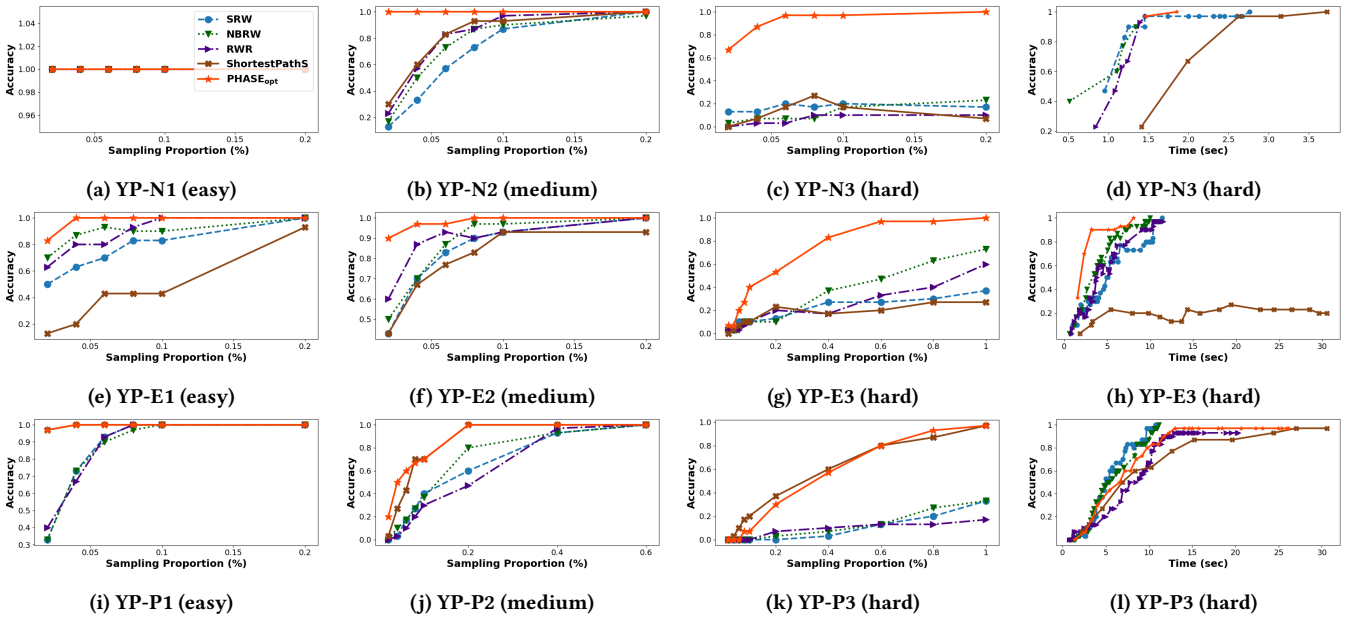
Figure 7: Comparison of the top five sampling methods for accuracy (a-c, e-g, i-k) and time-accuracy (d, h, l) performance across three node (a-d), three edge (e-h), and three path hypotheses (i-l) for DBLP.

sometimes shows competitive performance, possibly due to the high betweenness centrality of the concerned nodes, edges, or paths, which implies many shortest paths traversing them.

4.6 Scalability

In Table 4, the lowest execution time is in bold. First, the execution time increases with dataset size. Second, *RNS* generally requires the shortest execution time because it uniformly samples nodes. Our method, PHASE$_{opt}$, has varying time performance across datasets. Specifically, it ranks among the highest in execution time for Yelp, especially for path hypotheses. This is attributed to the limited node types that extend path extraction times. While for DBLP, its execution time ranks in the lowest five.

As PHASE$_{opt}$ can achieve high accuracy with a small B, it is unfair to compare the time efficiency at a fixed B. Instead, we plot accuracy versus execution time in subfigures d, h, and l of Figures 7 and 8 for DBLP and Yelp, respectively, for the top five samplers ranked by accuracy. Due to the space constraints, only the most



(a) YP-N1 (easy) (b) YP-N2 (medium) (c) YP-N3 (hard) (d) YP-N3 (hard)

(e) YP-E1 (easy) (f) YP-E2 (medium) (g) YP-E3 (hard) (h) YP-E3 (hard)

(i) YP-P1 (easy) (j) YP-P2 (medium) (k) YP-P3 (hard) (l) YP-P3 (hard)

Figure 8: Comparison of the top five sampling methods for accuracy (a-c, e-g, i-k) and time-accuracy (d, h, l) performance across three node (a-d), three edge (e-h), and three path hypotheses (i-l) for Yelp.

difficult hypothesis of each type is shown. We measure time and accuracy starting from $B = 1000$ in increments of 1000 until the accuracy reaches one, or time reaches 50s (DBLP) and 30s (Yelp). We find that PHASE$_{\text{opt}}$ consistently achieves high accuracy in the shortest time and with the least B. Also, the execution time does not grow exponentially. Moreover, when the path length increases, as shown in Figure 6c and 6d, PHASE$_{\text{opt}}$ achieves the highest accuracy in the shortest time.

5 RELATED WORK

There has been a lot of research interest on hypothesis testing [3, 8, 11, 12, 17, 25, 35] and graph sampling techniques [19–21]. In this section, we summarize the most representative works.

Hypothesis Testing on Graphs. Hypotheses on graphs can be categorized by their object of interest [3, 5, 36]. Tang et al. and Ghoshdastidar et al. [9, 10, 32] extend the one-sample problem into two-sample to test whether two groups of random graphs are obtained by the same generative model or with the same graph distribution. In [35], nodes are objects of interest, and one-sample hypothesis testing is used to detect conditional dependence between nodes in brain connectivity networks.

Our goal is closely related to works that focus on nodes as the object of interest. However, there are two major differences between our work and the existing ones. First, we enable more expressive hypotheses, including edge and path hypotheses, on graphs. Second, we leverage graph sampling methods to conduct hypothesis testing.

Sampling Methods on Graphs. Most sampling methods are designed to sample a representative subgraph to accurately estimate graph properties [16, 20, 22, 30]. Leskovec and Faloutsos [20] are the first to study this problem in real-world networks [20] by proposing sampling techniques to maintain degree distribution, clustering coefficient, and distribution of component sizes. Hübler et al. [16] propose the Metropolis-Hastings sampling method. Maiya and Berger-Wolf [22] define the community representativeness sample and propose a community structure expansion sampler. Later, many sampling methods are proposed to improve the efficiency and convergence rate of simple random walk [19, 21]. Besides representative sampling, some sampling methods are designed for specific tasks, such as graph compression [4], community detection [22], and graph visualization [26, 34]. However, none of them is designed for hypothesis testing on attributed graphs.

6 CONCLUSION

In this paper, we develop a framework for hypothesis testing on large attributed graphs, which accommodates 11 existing hypothesis-agnostic samplers and new hypothesis-aware samplers. We propose dedicated optimizations to speed up sampling while achieving high test significance and accuracy. We also demonstrate theoretically and empirically that our hypothesis-aware sampling achieves earlier convergence of the hypothesis estimator than other methods.

Our work opens several new directions in the area of hypothesis sampling on graphs. The first direction is to examine additional optimizations that make use of domain-specific information on the input graph. The second direction is to handle more expressive hypotheses and specify two-sample and multiple sample scenarios.

ACKNOWLEDGMENTS

Reynold Cheng, Yun Wang, and Chrysanthi Kosyfaki were supported by the University of Hong Kong (Project 109000579), the HKU Outstanding Research Student Supervisor Award 2022-23, and the HKU Faculty Exchange Award 2024 (Engineering).

REFERENCES

[1] 2015. Yelp Dataset. https://www.yelp.com/dataset.

[2] Lada A. Adamic, Rajan M. Lukose, Amit R. Puniyani, and Bernardo A. Huberman. 2001. Search in Power-Law Networks. *CoRR* cs.NI/0103016 (2001).

[3] Ery Arias-Castro and Nicolas Verzelen. 2014. Community detection in dense random networks. (2014).

[4] Maciej Besta, Simon Weber, Lukas Gianinazzi, Robert Gerstenberger, Andrey Ivanov, Yishai Oltchik, and Torsten Hoefler. 2019. Slim graph: practical lossy graph compression for approximate graph processing, storage, and analytics. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2019, Denver, Colorado, USA, November 17-19, 2019*. ACM, 35:1–35:25.

[5] Peter J. Bickel and Purnamrita Sarkar. 2013. Hypothesis Testing for Automated Community Detection in Networks. *CoRR* abs/1311.2694 (2013).

[6] Warren H Bondy and William Zlot. 1976. The standard error of the mean and the difference between means for finite populations. *The American Statistician* 30, 2 (1976), 96–97.

[7] William Gemmell Cochran. 1977. *Sampling techniques*. john wiley & sons.

[8] Darren P Croft, Joah R Madden, Daniel W Franks, and Richard James. 2011. Hypothesis testing in animal social networks. *Trends in ecology & evolution* 26, 10 (2011), 502–507.

[9] Debarghya Ghoshdastidar, Maurilio Gutzeit, Alexandra Carpentier, and Ulrike von Luxburg. 2017. Two-Sample Tests for Large Random Graphs Using Network Statistics. In *Proceedings of the 30th Conference on Learning Theory, COLT, Amsterdam, The Netherlands, 7-10 July (Proceedings of Machine Learning Research)*, Vol. 65. PMLR, 954–977.

[10] Debarghya Ghoshdastidar, Maurilio Gutzeit, Alexandra Carpentier, and Ulrike Von Luxburg. 2020. Two-sample hypothesis testing for inhomogeneous random graphs. (2020).

[11] Debarghya Ghoshdastidar and Ulrike von Luxburg. 2018. Practical Methods for Graph Two-Sample Testing. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. 3019–3028.

[12] Connor P. Gibbs, Bailey K. Fosdick, and James D. Wilson. 2022. ECoHeN: A Hypothesis Testing Framework for Extracting Communities from Heterogeneous Networks. *CoRR* abs/2212.10513 (2022).

[13] Minas Gjoka, Maciej Kurant, Carter T. Butts, and Athina Markopoulou. 2010. Walking in Facebook: A Case Study of Unbiased Sampling of OSNs. In *INFOCOM, 29th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 15-19 March, San Diego, CA, USA*. IEEE, 2498–2506.

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

[15] F. Maxwell Harper and Joseph A. Konstan. 2016. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4 (2016), 19:1–19:19.

[16] Christian Hübler, Hans-Peter Kriegel, Karsten M. Borgwardt, and Zoubin Ghahramani. 2008. Metropolis Algorithms for Representative Subgraph Sampling. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM 2008), December 15-19, 2008, Pisa, Italy*. IEEE Computer Society, 283–292.

[17] David R Hunter, Steven M Goodreau, and Mark S Handcock. 2008. Goodness of fit of social network models. *Journal of the american statistical association* 103, 481 (2008), 248–258.

[18] Vaishnavi Krishnamurthy, Michalis Faloutsos, Marek Chrobak, Li Lao, Jun-Hong Cui, and Allon G. Percus. 2005. Reducing Large Internet Topologies for Faster Simulations. In *NETWORKING: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communication Systems, 4th International IFIP-TC6 Networking Conference, Waterloo, Canada, May 2-6, Proceedings (Lecture Notes in Computer Science)*, Vol. 3462. Springer, 328–341.

[19] Chul-Ho Lee, Xin Xu, and Do Young Eun. 2012. Beyond random walk and metropolis-hastings samplers: why you should not backtrack for unbiased graph sampling. In *ACM SIGMETRICS/PERFORMANCE Joint International Conference on Measurement and Modeling of Computer Systems, SIGMETRICS, London, United Kingdom, June 11-15*. ACM, 319–330.

[20] Jure Leskovec and Christos Faloutsos. 2006. Sampling from large graphs. In *Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA, August 20-23*. ACM, 631–636.

[21] Yongkun Li, Zhiyong Wu, Shuai Lin, Hong Xie, Min Lv, Yinlong Xu, and John C. S. Lui. 2019. Walking with Perception: Efficient Random Walk Sampling via Common Neighbor Awareness. In *35th IEEE International Conference on Data Engineering, ICDE, Macao, China, April 8-11*. IEEE, 962–973.

[22] Arun S. Maiya and Tanya Y. Berger-Wolf. 2010. Sampling community structure. In *Proceedings of the 19th International Conference on World Wide Web, WWW, Raleigh, North Carolina, USA, April 26-30*. ACM, 701–710.

[23] Whitney K Newey and Daniel McFadden. 1994. Large sample estimation and hypothesis testing. *Handbook of econometrics* 4 (1994), 2111–2245.

[24] Jerzy Neyman. 1992. On the two different aspects of the representative method: the method of stratified sampling and the method of purposive selection. In *Breakthroughs in statistics: Methodology and distribution*. Springer, 123–150.

[25] Behrooz Omidvar-Tehrani, Sihem Amer-Yahia, and Ria Mae Borromeo. 2019. User group analytics: hypothesis generation and exploratory analysis of user data. *VLDB J.* 28, 2 (2019), 243–266.

[26] Davood Rafiei and Stephen Curial. 2005. Effectively Visualizing Large Networks Through Sampling. In *16th IEEE Visualization Conference, IEEE Vis 2005, Minneapolis, MN, USA, October 23-28, 2005, Proceedings*. IEEE Computer Society, 375–382.

[27] Alireza Rezvanian and Mohammad Reza Meybodi. 2015. Sampling social networks using shortest paths. *Physica A: Statistical Mechanics and its Applications* 424 (2015), 254–268.

[28] Bruno F. Ribeiro and Donald F. Towsley. 2010. Estimating and sampling graphs with multidimensional random walks. In *Proceedings of the 10th ACM SIGCOMM Internet Measurement Conference, IMC, Melbourne, Australia - November 1-3*. ACM, 390–403.

[29] Michael PH Stumpf, Carsten Wiuf, and Robert M May. 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.

[30] Daniel Stutzbach, Reza Rejaie, Nick G. Duffield, Subhabrata Sen, and Walter Willinger. 2009. On unbiased sampling for unstructured peer-to-peer networks. *IEEE/ACM Trans. Netw.* 17, 2 (2009), 377–390.

[31] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. ArnetMiner: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27*. ACM, 990–998.

[32] Minh Tang, Avanti Athreya, Daniel L Sussman, Vince Lyzinski, Youngser Park, and Carey E Priebe. 2017. A semiparametric two-sample hypothesis testing problem for random graphs. *Journal of Computational and Graphical Statistics* 26, 2 (2017), 344–354.

[33] Jaime Waters. 2015. Snowball sampling: A cautionary tale involving a study of older drug users. *International Journal of Social Research Methodology* 18, 4 (2015), 367–380.

[34] Yanhong Wu, Nan Cao, Daniel Archambault, Qiaomu Shen, Huamin Qu, and Weiwei Cui. 2017. Evaluation of Graph Sampling: A Visualization Perspective. *IEEE Trans. Vis. Comput. Graph.* 23, 1 (2017), 401–410.

[35] Yin Xia and Lexin Li. 2017. Hypothesis testing of matrix graph model with application to brain connectivity analysis. *Biometrics* 73, 3 (2017), 780–791.

[36] Yunpeng Zhao, Elizaveta Levina, and Ji Zhu. 2011. Community extraction for social networks. *Proceedings of the National Academy of Sciences* 108, 18 (2011), 7321–7326.