Task-driven sampling of attributed networks

Suhansanu Kumar
University of Illinois at Urbana-Champaign
Urbana, IL
skumar56@illinois.edu

Hari Sundaram
University of Illinois at Urbana-Champaign
Urbana, IL
hs1@illinois.edu

ABSTRACT

This paper introduces new techniques for sampling attributed networks to support standard Data Mining tasks. The problem is important for two reasons. First, it is commonplace to perform data mining tasks such as clustering and classification of network attributes (attributes of the nodes, including social media posts), on sampled graphs since real-world networks can be very large. And second, the early work on network samplers (e.g. ForestFire, Re-weighted Random Walk, Metropolis-Hastings Random Walk) focused on preserving structural properties of the network (e.g. degree distribution, diameter) in the sample. However, it is unclear if these data agnostic samplers tuned to preserve network structural properties would preserve salient characteristics of network content; preserving salient data characteristics is critical for clustering and classification tasks. There are three contributions of this paper. First, we introduce several data aware samplers based on Information Theoretic principles. Second, we carefully analyze data aware samplers with state of the art data agnostic samplers (which use only network structure to sample) for three different data mining tasks: data characterization, clustering and classification. Finally, our experimental results over large real-world datasets and synthetic benchmarks suggest a surprising result: there is no single sampler that is consistently the best across all tasks. We show that data aware samplers perform significantly better ($p < 0.05$) than data agnostic samplers on data coverage, clustering, classification tasks.

Keywords

Sampling; Networks; Data Mining; Clustering; Classifiers

1. INTRODUCTION

In this paper, we propose new sampling algorithms for attributed networks. By network attributes, we specifically mean content attributes such as gender, location etc. that are distinct from attributes of a node arising from network structure such as node degree, clustering coefficient to name a few.

The problem is important for several reasons. There is tremendous interest in performing data mining tasks such as classification [1], community discovery [2] as well as clustering of nodes into functional groups [3] using node content. However, real-world network datasets are extraordinarily large, either computational efficiency of the algorithm, or API limits of the social networks necessitates working with much smaller network samples. Most researchers use well known graph sampling methods such as snowball sampling, or stochastic samplers such as Random Walk, ForestFire [4] or Metropolis-Hastings Random Walk (MHRW) [5]. It is assumed that these samplers are "good enough" for data analysis. However, much of the early work on network sampling arose out a desire to understand the structure of the network, not to discover patterns in the content attributes of the nodes. Furthermore, while samplers such as MHRW have been shown to be asymptotically equivalent to uniform sampling the graph, however, finite sample statistics of MHRW reveal that for a finite sample, the probability of visiting a node is not uniform over the network.

To the best of our knowledge, this is the first paper to systematically analyze the effects of network sampling on data mining tasks that involve node content attributes. We specifically look at three tasks—data characterization, clustering and classification. Our contributions are as follows:

1. We propose several new *link-trace* samplers grounded in Information Theory. These "Information Expansion" samplers seek out new, previously unseen data samples rapidly covering the attribute range. We also introduce Pareto-optimal sampling methods that combine content-aware samplers with content-agnostic samplers such as MHRW and RW.

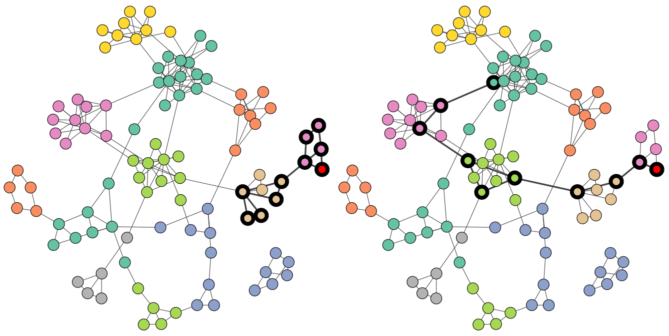
2. We characterize the bias of these Information theoretic samplers, and prove that they tends to a uniform distribution over the sampled set and its neighborhood. Furthermore, we can show that for highly assortative networks, these information-theoretic samplers will prefer to seek out new clusters over existing clusters, thus rapidly covering the data space.

A surprising result: A systematic analysis over diverse, large real-world datasets and extensive experiments over synthetic attributed networks reveals that *no single sampler is optimal for all data mining tasks.* In all three cases—characterization, clustering and classification—content aware samplers outperform baselines (BFS, RW, MHRW) in a statistically sig-

WOODSTOCK '97 El Paso, Texas USA

© 2016 ACM. ISBN 123-4567-24-567/08/06...$15.00

DOI: 10.475/123_4



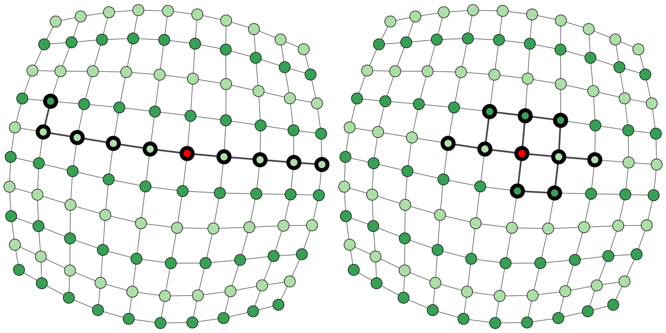
(a) a MHRW run (b) an IXS run

Figure 1: The Figure shows a network where the different colors represent the different attribute values; sampled nodes with darker concentric rings; the target sample size is 10%. Subplot (a) shows MHRW, a random walk based sampler, getting stuck in a local part of the network, due to its bias. Subplot (b) shows data aware sampler IXS overcoming this bottleneck due to its bias for new information.

nificant manner ($p < 0.05$). For data characterization task only, uniform random sampling has the best raw performance ($K - S$ statistic). However, this is statistically indistinguishable from content aware link-trace sampling. For clustering and classification tasks, content aware samplers are far superior ($p < 0.05$) to all baselines including uniform, with a 25% best case improvement.

We show via a stylized example the differences between two link-trace samplers—MHRW and Information eXpansion Sampling (IXS)—for a finite sample from an attributed network. In Figure 1, we show a network with a strong community structure and having a single discrete attribute; the different colors in the graph refer to different attribute values. The two sub-figures show a single trace of size equal to 10% of the network size, of two algorithms (MHRW, Information eXpansion Sampler,) starting from the same seed node (marked in red). The sampled nodes are marked with a dark black ring, and the edges of the induced subgraph are colored black. As can be seen from Figure 1, MHRW gets "stuck" in a small section of the network even though it has an asymptotically optimal performance characteristic. For a small sample size, MHRW has a known bias towards low-degree nodes. Notice that IXS with its bias towards capturing new attribute values is much more efficient at covering the attribute space. In a similar vein illustrated through Figure 2, we can show that IXS is superior to XS [6] a data-agnostic sampler that performs well with networks with community structure, for dis-assortative networks with poor community structure. In summary, our proposed Information eXpansion Sampling algorithm expands rapidly in the data space when there are attributes to be discovered, but more like a random walker if the information in the network neighborhood of the sample fails to provide guidance.

The rest of this paper is organized as follows. In the next section we formally define the sampling problem. In Section 3, we discuss data-agnostic and data-aware papers and introduce our information expansion based samplers. In the three following sections, we present results for synthetic and real-world datasets for baseline and our data-aware samplers for data characterization, clustering and classification tasks. In Section 7, we discuss prior work and in Section 8, we discuss limitations. Finally, we present our conclusions



(a) a XS run (b) an IXS run

Figure 2: The Figure shows a network where the two shades of green represents the two different attributes of the nodes. Subplot (a) shows expansion sampling (XS) behaves poorly when the attribute color has low assortativity of 0. Subplot (b) shows data awareness while sampling can alleviate this problem. The two shades of green represent the different attribute values.

2. PRELIMINARIES

In this section, we shall first define the notation used in the paper, then we shall formally define the task-driven (or purposeful) sampling problem in Section 2.1. We shall conclude the section with a discussion of the real-world datasets used as well as the mechanisms to create realistic attributed datasets.

Table 1: Notation used in the paper

Symbol	Definition
G	Network, $G = (V, E)$
V	Vertices of the network
E	Edges of the network, $E \subseteq V \times V$
v	vertex(node) in network
d_v	degree of node v in network
d_V	mean degree of nodes $v \in V$: $d_V = \frac{1}{\lvert V\rvert} \sum_v d_v$
$A(v)$	Attribute vector or the data on node v
$\mathbb{S}$	Sampled nodes
$N(\mathbb{S})$	Frontier node set or neighborhood of set $\mathbb{S}$ $\{w \in V \setminus \mathbb{S} : \exists v \in \mathbb{S} : (v, w) \in E\}$
$\Delta(v)$	Unexplored neighbors of a node v $\{w \in N(v) \setminus (\mathbb{S} \cup N(\mathbb{S}))\}$
L	Set of categorical attributes in G
M	Set of continuous attributes in G
$\mathbb{C}$	Set of data clusters in G.

2.1 Problem Statement

Assume that we have a graph $G = (V, E)$, where each node has m content attributes (e.g. gender, location, etc.) and that we have a task F that performs operations using an attributed graph as input. The task F produces an output: a data characterization in form of a scalar (e.g. mean of an attribute), or a vector (e.g. distribution of gender); a mapping (e.g. assignment of each node to a cluster); a function (e.g. a classifier that operates on further input).

Further assume that the input to F is a sample $\mathbb{S}$ of size $k \ll |V|$. The sample $\mathbb{S}$ can be obtained through a variety of ways, including for example, random sampling of nodes). However, in practice random access to the nodes is rare. Instead, most network sampling mechanisms are *link-trace* samplers. We define link trace sampling in a manner similar to [6] as follows. Given an integer k and an initial seed node $v \in V$ to which the $\mathbb{S}$ is initialized, a link trace sampler L adds nodes v to $\mathbb{S}$ such that there exists a node $w \in \mathbb{S}$ where $(w, v) \in E$. The sampler stops when $|\mathbb{S}| = k$.

Link-trace samplers yield connected components since each new addition must lie in $N(\mathbb{S})$, the neighborhood of $\mathbb{S}$. We ran the link-trace samplers on largest component of the graph leading to a sample collection of k nodes from the $|V|$ nodes of the underlying network.

This sample $\mathbb{S}$ is associated with an induced subgraph $G'_k = (V_k, E_k)$. Thus for a given sample size k, seed nodes θ and a task F, the goal is to find an optimal link-trace sampler L^* such that,

$$L^* = \arg\min_L \mathbb{E}_\theta D\left(F(G), F(G'_k; L, \theta)\right) \qquad (1)$$

The function D measure the distance between the outputs for task F in the ideal case with the entire graph G as input against the case when the sampled graph G'_k is used as input. The graph G'_k is parameterized by the sampler L and the seed set θ. The distance measure is task dependent: D could be just the absolute difference in values, say when F is computing the mean of an attribute, the $K - S$ statistic in the case when F computes a distribution, or Normalized Mutual Information when F performs clustering.

There may be two sources of randomness involved in sampling, depending on the type of sampler. The first source is the location of the seed set and the second source of randomness may be the sampler itself. In the case of stochastic samplers like Random-Walk, for the same seed, every run of the algorithm will produce a different sample. The expectation $\mathbb{E}$ is over both sources of information, although Equation (1) only refers to the expectation over θ. Thus, Equation (1) simply says that we should select the link trace sampler with minimum distance to the ideal case, averaged over different seed sets and over different link traces.

2.2 Datasets

In this section, we discuss the real world datasets and generators to synthesize attributed network datasets.

We consider three real world datasets: Facebook, the large patent citation network and the well studied Enron communication network. The three networks significantly differ in size, and in key network parameters: degree distribution, diameter and clustering coefficient. See Table 2 for a summary of network characteristics. The networks also differ significantly in terms of attribute cardinality, attribute type (discrete vs. continuous attributes), data skew and assortivity (e.g. Patent category is most assortative with value 0.64). We perform our experiments on the largest component of the pre-processed network.

Table 2 presents structural properties of the three real world networks. The Facebook network [7] is a friendship network. This network only has discrete attributes. The patent network [8] is the citation network of all patents granted by the US from 1963 till 1999. The patent network has both discrete and continuous attributes. In the Enron network [9] each node is an individual and edges represent communication between the corresponding individuals. Enron has only continuous attributes described in Table 3.

Table 2: Network statistics. N: number of nodes, E: number of edges, DS: number of discrete attributes, CT: number of continuous attributes, d_V:average node degree, CC: local clustering coefficient, DI: diameter

Networks	N	E	DS	CT	d_V	CC	DI
Facebook	4,039	88,234	3	0	43.69	0.62	8
Enron	36,692	183,831	0	7	10.02	0.72	13
Patent	2,738,012	13,963,839	4	3	10.20	0.09	83

Table 3: Attribute statistics for three networks: Facebook, Patent and Enron. Type: Continuous(CT) or Discrete(DS) CD : attribute cardinality; SW: Skew; CV: attribute coverage over the nodes in the network; AS: assortativity The coverage of continous attribute is measured as their coverage of 10 log spaced bins. Due to very dense categorization in the original dataset, we use the "subcat" from the original dataset as our category and classes of the patent classes as subcategories.

Attribute	T	CD	SW	CV	AS
Facebook					
gender	DS	2	0.06	97.92	0.09
locale	DS	10	0.69	98.56	0.34
education type	DS	3	0.52	74.60	0.08
Patent					
category	DS	36	0.06	100.0	0.64
sub-category	DS	58	0.52	100.0	-0.02
assignee type	DS	7	0.40	100.0	0.25
country of origin	DS	69	0.70	100.0	0.20
citations made	CT	[0, 254]	1.00	100.0	-0.00
citations received	CT	[0, 2142]	0.31	100.0	0.06
claims made	CT	[0, 263]	1.00	100.0	0.04
Enron					
AvgContentLength	CT	[0, 9296]	1.00	100.0	0.02
AvgContentReplyCount	CT	[0, 1238]	0.92	100.0	0.10
AvgNumberTo	CT	[0, 2653]	0.95	100.0	-0.03
AvgContentForwarding	CT	[0, 1004]	0.85	100.0	0.10
AvgNumberCc	CT	[0, 1827]	0.99	100.0	0.04
AvgRangeBetween2Mails	CT	[0, 10313]	1.00	100.0	-0.00

We considered other datasets (e.g. those in [2]), but many datasets have significant number of nodes with missing attributes; this creates a confound since we don't know if the missing values are due to improper sampling. Hence we've used real-world network datasets with the fewest missing attributes, that are described in detail in Table 3.

There are three elements to synthetic network generation: the network structure, the attributes and the relationship between attributes and network structure.

For the network generation, we use the Lancichinetti-Fortunato-Radicchi (LFR) [10] algorithm to generate artificial networks of size $N = 1000$, with mixing coefficient $\mu = 0.1$ that resemble real world networks. with strong community structure. Such networks are referred to as LFR(μ =0.1), in the rest of the paper.

There are three essential data characteristics: skew, purity and assortivity. Assume that we have a single discrete attribute that takes on k values; this discussion is easily extended to multiple discrete attributes and to continuous attributes. Now, in the discrete one dimensional case, all data points sharing the same distinct attribute value will be grouped together into one cluster, C_k.

The data skew $s(\mathbb{C})$ of a set of clusters $\mathbb{C} = \{C_1, C_2, \ldots, C_k\}$ is defined in terms of Shannon entropy $H(\mathbb{C})$ over the cluster size. We shall use three discrete skew values of (low, medium, high) when generating the attributed network. The purity p of the data refers to the separability of the data clusters; this is parameterized by the standard deviation of the continuous variable and is easily extended to discrete [11] We use two extreme purity values to synthesize the network. Assortativity measures the degree to which nodes similar attributes are connected to each other that is significantly different from random matches [12]. We use two levels (low, high) of assortivity, where the low case corresponds to random assignments of attributes and the high case corresponds to $a \approx 1$. We note that negative assortivity is hard to achieve when the number of attribute values is large since cross attribute edges are similar to random matches.

Finally, we assign the synthesized data to the nodes in the network according to the specified assortative value through a label propagation algorithm that terminates when the target assortivity is achieved. We use the principle of *swapping* and *propagation* to map the data onto the network. In the swapping technique, an extreme (high, low) assortative distribution over network is first generated using the community detection and approximate k-coloring problem respectively. Randomly swapping categorical attribute between pair of vertices causes assortativity to tend to zero which is stopped when the target assortativity is achieved. The propagation algorithm propagates the same category with a proportional high probability if the target assortativity is high and vice versa.

In this Section, we defined all the symbols used in the paper, and formally defined the task-driven sampling problem; we specifically focus on link-trace samplers in this paper. Then we discussed the real-world and synthetic datasets used in this paper. In the next Section, we discuss different network sampling methodologies.

3. HOW TO SAMPLE

Let us denote the sample collected from the network as $\mathbb{S}$. The set of nodes that have at least one neighbor in $\mathbb{S}$ is the set of frontier nodes, denoted as $N(\mathbb{S})$. For some sampling algorithms, we shall be interested in Δv, the neighbors of node v that do not belong to $\mathbb{S} \cup N(\mathbb{S})$.

The set of sampling algorithms considered in this paper fall under the description of link-trace sampling. In such sampling schemes, the next node v selected for inclusion in the set $\mathbb{S}$ is a neighbor of at least one node in $\mathbb{S}$. In other words, $v \in N(\mathbb{S})$. We keep adding nodes until we have collected the number of nodes $k = |\mathbb{S}|$ that we desire. Typically $k \ll n$, where n is the number of nodes in the graph.

Link trace sampling is often the only practical option in many real-world scenarios (e.g. crawling a social network such as Twitter or Facebook), where we do not have random access to nodes in the graph. Such random access may be restricted by the social network. Furthermore, even if there is a unique numeric `id` associated with each individual, the `id`s may not be sequential requiring us to do costly rejection sampling [13]. The inability to randomly access each node in the graph implies that uniform sampling, which is the gold standard for sampling content, is largely infeasible on a graph.

We can broadly categorize sampling algorithms as either data agnostic, or data aware. The key difference lies in if the sampling algorithm uses the node attributes to decide whether to include it in the sample. We shall discuss data agnostic sampling next, followed by a discussion of data aware samplers in Section 3.2. In Section 3.3 we discuss the effects of side information including knowing the prior and the number of data clusters, on the sampling problem. Finally, we conclude this section, by showing how to combine the two types of sampling algorithms into Pareto optimal sampling algorithms; we discuss this issue in Section 3.4.

3.1 Data Agnostic

Data agnostic algorithms, including breadth first search (BFS), Forest Fire [4], Metropolis Hastings Random Walk (MHRW) [14], Random Walk (RW) and Expansion Sampling (XS) [6], do not use the content of any node to construct $\mathbb{S}$. For example, consider a graph $G = (V, E)$ that represents a typical social network, where the vertices V represents the individuals and where the edges E represents a relationship between individuals. In such a graph, one would expect each node in the graph to be associated with an array of attributes, including age, gender, interests. Data agnostic sampling algorithms ignore such node attributes when determining which nodes to add to the sample set $\mathbb{S}$.

The reasons why these well known sampling algorithms ignore node attributes vary. Some of the early sampling algorithms such as Forest Fire [4] were explicitly designed to preserve graph structural properties in the sub-graph such as degree distribution, densification etc. The other reason is that probabilistic samplers including MHRW and RW have important asymptotic properties that may be exploited. The Metropolis-Hasting sampling algorithm is actually asymptotically optimal for content since the probability of visiting any node in the graph is the same; thus in principle, using MHRW should be equivalent to uniform sampling of the graph. However, as we discuss below, the *finite sample* behavior of MHRW is less than optimal. Next, we introduce the well known data agnostic sampling algorithms in more detail.

In snowball sampling, a small seed set of vertices (θ) is used to initiate collection of data through BFS. While snowball sampling is computationally efficient, it is biased towards high degree nodes [15] and is also sensitive the selection of the seed nodes [6].

[4] proposed Forest Fire, which explores a subset of a node's neighbors according to a "burning probability" p_f; at $p_f = 1$, Forest Fire is identical with BFS. At each iteration, the subset of the neighbors of the current node v are chosen using a geometric distribution. While Forest Fire is superior to BFS, it also suffers from a degree bias [15].

[6] proposed expansion sampling (XS), motivated by expander graphs and greedily expands in the direction of the largest unexplored region. That is, the candidate node v^* selected to be added to $\mathbb{S}$ is chosen as follows:

$$v^* = \underset{v \in N(\mathbb{S})}{\arg\max} \, |N(v) - (\mathbb{S} \cup N(\mathbb{S}))| \qquad (2)$$

In other words, Equation (2) finds that node v^* that has the largest number of neighbors outside of the set $\mathbb{S} \cup N(\mathbb{S})$. [6] suggest that XS is very stable with respect to seed selection. While XS will rapidly discover homogeneous communities, since the sampling algorithm is data agnostic, it does less well over disassortative networks.

Re-weighted Random Walk sampling (RW) is a variant of the classic Random Walk algorithm, re-weighted to provide a better estimate of the content distribution. The reweighting is necessary since the random walk algorithm is biased towards high degree nodes. Notice that the stationary probability π_v of visiting a node v is proportional to the node degree. Hence the label associated with the node v of degree d_v is discounted by its degree; attribute probabilities are estimated through the Hansen-Hurwitz estimator [16] to develop an unbiased estimate of the content. Assuming an attribute A can take values (A_1, A_2, ... A_m), the unbiased probability distribution ($\widetilde{p}$) estimate of any discrete attribute A from a RW sampled collection is :

$$\widetilde{p}(A_i) = \frac{\sum_{u \in A_i} 1/d_u}{\sum_{u \in V} 1/d_u} \tag{3}$$

Similarly, we can use kernel density estimators for continuous attributes. Metropolis-Hasting random walk sampling (MHRW) alters the transition probabilities between pairs of nodes such that the stationary probability of visiting each node is the same. It sets the transition probability from node v to node w to be $\min(1, \frac{d_v}{d_w})$. While the asymptotic performance of MHRW is ideal in the sense that it simulates uniform sampling on the graph, it suffers from less than ideal finite sample behavior. Poor finite sample behavior is especially observable on graphs with strong community structure causing the MHRW to get stuck in a local community. MHRW typically requires sample sizes of $O(n)$, where n is the number of nodes in the graph, to achieve the stationary distribution. However since our sample sizes for large graphs will likely be extremely small (i.e. $|\mathbb{S}| \ll n$), the finite sample performance of MHRW becomes very important.

3.2 Data Aware

Data aware samplers use node content to determine the sample set $\mathbb{S}$. These samplers determine the next node v to be added to the current sample set $\mathbb{S}$, by checking the content of this node against the content of the nodes in the current sample. At any point in time, we have the set $\mathbb{S}$, comprising nodes in the current sample set, $N(\mathbb{S})$, the set of all frontier nodes who have at least one neighbor in $\mathbb{S}$. At each step, we shall add to $\mathbb{S}$, one optimal node $v \in N(\mathbb{S})$. We shall assume that for each $vinN(\mathbb{S})$, we shall have access to the content of the neighbors of v. This similar in spirit to Expansion Sampling proposed by [6], where they assume that for each node $v \in N(\mathbb{S})$, we know the neighbors of v, that do not belong to $\mathbb{S}$. We call the set of neighbors of v, that do not belong to $\mathbb{S}$, the candidate set for node v and shall designate it with Δv.

The remainder of this section organized as follows. In the next section, we shall introduce the idea of surprise that is grounded in Information Theory, and develop several algorithms that incorporate this idea to sample network data. Then, in Section 3.2.2, we shall introduce the idea that extremal points, ones that are far away from all the points in the current sample, are interesting and we shall present an sampling algorithm that uses this idea. Finally in Section 3.2.3, we show develop a data-aware version of MHRW and analyze its shortcomings.

3.2.1 Surprise Based Sampling

Surprise based samplers compute the extent to which the candidate set Δv can be predicted by what is known thus far: the set $\mathbb{S} \cup N(\mathbb{S})$.

In Information Expansion Sampling (IXS), surprise $I_{\Delta v}$ of a candidate set Δv (with respect to $\mathbb{S}$) is computed as follows:

$$\begin{aligned} I_{\Delta v} =& \frac{-\ln P(\Delta v|\mathbb{S})}{|\Delta v|} \\ =& \sum_{i=1}^{k} p_{\Delta v}(i) \ln p_{\mathbb{S}}(i) \end{aligned} \tag{4}$$

where, k is the number of distinct attribute values, $p_{\Delta v}(i)$ is the probability of attribute value i in the candidate set Δv, and $p_{\mathbb{S}}(i)$ is the probability of the attribute value i in the sample set $\mathbb{S}$.

Selection of nodes from the neighborhood set $N(\mathbb{S})$ in a manner that maximizes surprise Equation (4) creates a bias in IXS, which we illustrate next.

Lemma 1. *Given a sample set $\mathbb{S}$, Information Expansion Sampling for a discrete attribute with two values is biased towards a uniform distribution over the attribute values in $\mathbb{S}$.*

Proof. Let us assume that the two attribute values $\{1, 2\}$ occur in the set $\mathbb{S}$ with probabilities p and $1 - p$ with $p < 1/2$. Further assume that in the sample candidate set Δv, attribute value 1 occurs with probability x and attribute value 2 occurs with probability $1 - x$. Thus the attribute value with lower probability occurs in the candidate set Δv occurs with probability x.

The surprise $I_{\Delta v}$ associated with the set Δv is defined as:

$$\begin{aligned} I_{\Delta v} =& -x \ln p - (1 - x) \ln(1 - p) \\ =& -\ln(1 - p) + x \ln \frac{1 - p}{p} \end{aligned} \tag{5}$$

Since $I_{\Delta v}$ is linear in x and since $p < 1/2$, IXS will pick that node $v^* \in N(\mathbb{S})$ with the largest value of x to maximize $I_{\Delta v}$. In other words, IXS will pick the node v^* with the largest fraction of least probable attribute.

If $x > p$, then the entropy of the updated sample set $\mathbb{S} \cup v$ is guaranteed to increase due to the fact that the entropy $H(p)$ of the sample $\mathbb{S}$ is concave in p. Note that $H(p)$ is maximized when the distribution is uniform. □

There are two important things to note with Lemma 1. First, despite the bias towards the uniform distribution, when $|\mathbb{S}| = O(n)$, where n is the number of nodes of the graph, the distribution of the attributes in the sample $\mathbb{S}$ will tend to the underlying distribution of the attribute in the graph, which may not be uniform. Thus Lemma 1 is primarily useful when $|\mathbb{S}| \ll n$, which is indeed the case of interest.

IXS rapidly expands to discover unseen attribute values. This is driven by the observation that unseen attribute values (i.e. $p_{\mathbb{S}}(i) = 0$) in $\mathbb{S}$ will cause Equation (4) to diverge. Thus IXS first rapidly covers the attribute space, and then favors the uniform distribution of attributes in $\mathbb{S}$. While IXS is indepndent of the size of $\mathbb{S}$ and the size of Δv, the rate of change in sample entropy $H(p)$ does depend on size.

One can change the definition of surprise to be based on the Kullback-Liebler divergence between the attribute distribution in the sample $\mathbb{S}$ and the candidate set Δv. We term this samplers based on KL divergence as KL-information expansion sampling (kl-IXS).

$$I_{\Delta v} = \sum_{i}^{k} p_{\Delta v}(i) \ln \frac{p_{\Delta v}(i)}{p_{\mathbb{S}}(i)} \qquad (6)$$

Since kl-IXS behaves very similarly to IXS in our experiments, and thus we shall omit results pertaining to kl-IXS while presenting our results.

Thus far, we've discussed surprise for categorical variables. We can extend the notion of surprise to continuous variables by simply discretizing the continuous variables and then computing the surprise using Equation (4) with the discretized values. Another option is to estimate the underlying density, say using a mixture model and then assign the nearest mixture to the feature vector corresponding to each node in the candidate set Δv. Similarly, one could assign each feature for each node in $\mathbb{S}$ to one of the mixtures. Now, we can again compute surprise as before with the discrete case. Information expansion samplers that use both continuous and discrete variables are termed as Hybrid IXS (H-IXS). In general, hybrid samplers do better than Information expansion samplers and extremal point sampling discussed next.

3.2.2 Extremal Point Sampling

A node that is at a large distance, in terms of its features, from the all the current nodes would be surprising. We use this idea, which we term extremal point sampling (ExP), to identify surprising nodes for nodes with continuous features. In ExP, we rank the candidate nodes in terms of their average distance to all the nodes in the sample set $\mathbb{S}$. The node with the highest rank is then added to the sample set $\mathbb{S}$. While we could use many different distance measures, we choose to use the standard Euclidean distance. Mahanalobis distance with its covariance correction would be the ideal Euclidean distance choice, but is not used due to the difficulty in developing a stable estimate for the covariance matrix with a small sample.

3.2.3 Surprise based MHRW

Could we make MHRW data aware? One possibility is to couple the surprise for each node $v \in N(i)$, where $N(i)$ is the neighborhood of i, the node where the MHRW sampler is at present. We could define the probability $\hat{p}_{i,v}$ of jumping from node i to node v as:

$$\hat{p}_{i,v} \propto p_{i,v} I_{\Delta v} \qquad (7)$$

where, $I_{\Delta v}$ is the surprise with respect to the sample set $\mathbb{S}$ and $p_{i,v}$ is the probability of transitioning to v from i in the original MHRW sampler.

This approach has intuitive appeal since it appears to combine the best ideas from data agnostic samplers with that of surprise based data aware samplers; in addition unlike IXS or H-IXS, it is not a deterministic algorithm. The challenge is that Equation (7) changes the stationary distribution of the sampler—we are no longer guaranteed uniform stationary distribution over the graph nodes. Such data aware MHRW algorithms are also harder to analyze since the process is no longer first order Markov. Regardless, the idea that one could combine data aware and data agnostic samplers has obvious appeal and we shall consider this idea in more detail in Section 3.4. We propose a simplistic combination sampler from IXS and MHRW that chooses non-deterministically with equal probability to sample from either of the strategies. We call this combination sampler as IXS and MHRW or I&M.

3.3 Sampling with Side Information

All the algorithms discussed thus far assume no prior knowledge of the structural characteristics of the network or anything about the distribution of the data. However, often, we may have a rough idea of either the properties of the network, say the skewness of the degree distribution or of the attribute values (e.g. most of the Twitter users are from the U.S.). How should we incorporate this side information into the sampling process?

We have explored this idea with respect to sampling content properties of the network. For example, assume that we have access via an oracle, to the underlying attribute distribution p over the entire network. Then, one could simply use the earlier surprise based criteria to add additional nodes to the sample $\mathbb{S}$, except that instead of using $P_{\mathbb{S}}(i)$ in Equation (4), we use p_i. Notice that p is a constant while $P_{\mathbb{S}}$ is variable. This change will ensure that samples collected in $\mathbb{S}$ will have an attribute distribution that matches p.

When the prior p is unavailable, one could proceed as follows. We first MHRW till the sample statistic (say the distribution mean) converges via Gilman Ruben or Gweeke statistic [13] and then estimate $\hat{p}$, the sample attribute value distribution. Then, we proceed as earlier and use $\hat{p}$ in the surprise calculation.

Another approach is to incorporate knowledge of the underlying data clusters, or data classes, which may be known (e.g. if all residents in a U.S. state form a class, then there are 50 classes). Assume then, that we know k the number of data clusters. We can combine the IXS (for categorical data) and the ExP samplers (for the continuous feature vectors) as follows. From the sampled set $\mathbb{S}$, we construct k data clusters with different centers using the continuous data. Then, we assign each node in $\mathbb{S}$ and in Δv to the nearest cluster center. Now, we can compute surprise as earlier based on the cluster `id` distribution of Δv in conjunction with the surprise of Δv with respect to the distribution of categorical attributes in $\mathbb{S}$.

On the other hand if the content distribution of the original network is known from practice or approximation [5], we leverage the variable neighbourhood search (VNS) approach to select nodes sequentially that preserve the known prior content distribution in the hope of gaining a better representative sample. In other words we select the node $v*$ in $N(S)$ which minimizes KS statistic (D-stat) of the sample and original distribution.

$$v* = argmin_{v \in N(S)} D(A(V_S), A(V)) \qquad (8)$$

3.4 Pareto-Optimal Sampling

It would be ideal to develop a sampler that could preserve the properties of the network content as well as structural properties of the network. We do this via a Pareto-optimal sampler that combines MHRW based sampling with surprise based sampling. Assume that we have a sample set $\mathbb{S}$. Then, $\forall v \in N(\mathbb{S})$, we can compute two numbers: the probability of reaching v from $\mathbb{S}$ via MHRW and $I_{\Delta v}$ the surprise due to node v. Thus we can compute the Pareto-optimal frontier using all $v \in N(\mathbb{S})$ and choose the node from the frontier that best suits our bias (equal weight to structural properties and to content). We provide results of a pareto-combination of IXS ($I_{\Delta v}$) and XS ($|\Delta v|$) called as pIX (pareto-IXS-XS) and another pareto combination of IXS ($I_{\Delta v}$) and MHRW (under independece assumption of transition probabilities, $\frac{\sum_{u \in S} min(1/d_v, 1/d_u)}{\sum_{w \in N(S)} \sum_{u \in S} min(1/d_w, 1/d_u)}$) called as pIM (pareto-IXS-MHRW).

In this section we discussed data agnostic and data aware sampling schemes. The main idea behind data aware sampling algorithms is to preserve node content properties including content distribution in the sample. We introduced the idea of surprise, grounded in Information Theory, as a metric to develop sampling schemes. We also showed how to incorporate side information into the sampler as well as developing a Pareto-optimal sampling scheme. Next, we evaluate these sampling schemes on characterizing the node content.

4. DATA CHARACTERIZATION

In this section, we discuss how samplers preserve the statistical characteristics of the attributed graph—including statistical properties of the network structure (e.g. degree distribution), distributions of attributes and joint network-data properties including assortivity which relates network structure to node attributes. We shall conclude this section by presenting experimental results that help to compare different samplers.

4.1 Properties

We study the three salient properties of an attributed network: network structure, content structure and network-content relationship.

Network properties: We shall evaluate different samplers with respect their fidelity towards several key network properties. We chose to use a subset of the properties—*degree distribution, clustering coefficient, diameter*—evaluated in [4, 6], as these three properties are widely used to characterize networks.

Content Characteristics: We would like samplers to preserve essential aspects of the node content. By the phrase "node content," we refer to the attributes such as "gender=female," "ethnicity=asian"; we are using the word "content" to refer to all nodal attributes that are not derived from structural properties of the graph, including degree, clustering coefficient etc. The usage of "node content," over the more common "node data" will clearly distinguish attributes that are not derived from network structure, from those that are related to structure. We use the familiar Kolmogorov-Smirov(K-S) statistics to compute the distance between two sample distribution and the underlying ground truth distribution. For categorical attributes, K-S statistics reduces to variational distance. Another key content characteristics is *attribute coverage* which is defined as the ratio of unique attribute values in the sample to the attribute values in the underlying content. For the continuous attributes, we capture the range corresponding to cardinality via logarithmic binning. We also discuss preservation of *content clusters and classes* in detail in Section 5.

Joint Network-Content Relationships : Network structure and node content are often correlated and hence preserving this correlation is also important. For example, the correlation can arise due to homophily [17] when friendships form when like minded individuals seek out each other. Assortativity [12] is one of the widely used relationship metric to measure this correlation.

4.2 Experimental setup

We evaluate the suite of samplers discussed in the previous Section 3 under the following experimental conditions.

We choose three attributed networks: Facebook, Patent and Enron for content analysis. The choice of the network was based on the availability of content, that was ensured to be an inherent and immutable node property with a high degree of coverage, like "gender" of a user in Facebook network and "category" of a patent in Patent network which are independent of network structure. The only exception considered was "in-citations" in Patent network which acts as a pseudo of the quality of patent. Most of the attributed real-world networks such as Google+, Twitter, etc. were unsuitable for our use since they suffer from content sparsity perhaps due to high degree of noise or poor sampling. However, we recognize that future analysis must address this issue of noise and sparsity. From these networks, we choose contrasting attributes to cover a wide range of data characteristics such as cardinality/range, skew, assortativity and purity.

Evaluation: We evaluate the results of using the mean average distance defined as,

$$\bar{D}(k) = \frac{\sum_{k=1}^{K} \sum_{s=1}^{S} D(F(G), F(G'_{\alpha})}{K \times S} \tag{9}$$

where, $\alpha = k \times N/100$, K is the target sample size percentage, S is the number of simulations, N is the size of the network and *distance* metric is variable that depending on the task could be defined as KS stats between distributions, normalized mutual information between cluster partitioning and so on. Size of sampled nodes is usually orders of magnitude smaller than the original size. Therefore for our experiments, we averaged the distance for sample sizes is 1%, 2%, 3%, up till 10%(=K) of the original network. At each sample size, we varied the seed nodes by randomly selecting a different starting node for 100 times(=S). Detailed analysis on all of the network attributes is provided in the extended report [18]. The missing values in Tables 4 to 7 have missing values (dashed) for some of the samplers like IXS that are specifically defined for discrete content and therefore not applicable to continuous content and vice versa.

4.3 Experimental results

Table 4 shows the mean average KS statistics for three continuous and three discrete attributes in the three real world networks. The samplers are divided into three classes based on their sampling methodology and behavior. Observe that UNI is the best sampler among all samplers for capturing content distribution. It creates an unbiased sample estimate of the content distribution making it suitable for all attribute types. The content-aware samplers statistically significantly outperforms the content-agnostic sampler on nearly every attribute.

We note some interesting observation about individual samplers and their performance under special scenarios. We notice that VNS even though has prior content distribution is outperformed by UNI due to the network structure limitation. XS does remarkably well when the content is high clustered like in "locale". On the other hand, it does significantly worse when the assortativity is low and content is not clustered across the network, e.g. "subcategory" attribute. We find similar results as reported in [13] about re-weighted RW performing better than MHRW. Thus, even though MHRW and RW sample uniformly at random, their performance departs from the theory at low sample sizes significantly.

Table 5 shows ability of the samplers at exploring different attribute values. Logarithmic binning (#bins = 10) is per-

Table 4: The data distribution goal is to have least-possible mean average KS statistic (0) between original and sampled content distribution. UNI is empirically and analytically the best sampler. Symbol * indicates that content-aware samplers outperforms a content-agnostic baseline by 95% statistical confidence at 5% sample size. The boldface represents the best performing sampler. Abbreviations: # *forward*: average emails forwarded by the user, *time*: average time between two emails

	Facebook		*Patent*		*Enron*	
Samplers	gender	locale	subcategory	in-citation	# forward	time
BFS	0.082*	0.177*	0.071*	0.066*	0.379*	0.857*
RW	0.066	0.167*	0.068*	0.072*	0.443*	0.895*
MHRW	0.079*	0.178*	0.063*	0.075*	0.339*	0.872*
FF	0.081*	0.172*	0.076	0.158*	0.433*	0.895*
XS	0.057	0.053	0.208	0.428*	0.414*	0.847*
UNI	**0.034**	**0.046**	0.027	**0.010**	0.126*	**0.729**
VNS	0.043	0.083	**0.022**	0.044	-	-
IXS	0.072	0.161	0.058	0.035	-	-
ExP	-	-	0.166	0.201	**0.117**	0.736
I&M	0.073	0.166	0.049	0.036	-	-
pBM	0.073	0.165	0.050	0.046	-	-
pBX	0.073	0.105	0.197	0.429	-	-
Mean	0.066	0.134	0.088	0.133	0.321	0.833

formed to capture the range of continuous attributes because all continuous attributes in our dataset were observed to be exponentially distributed. We observe that the unbiased UNI sampler performs pretty well on real world attribute coverage as well. However, it gets outperformed by the content-aware samplers like IXS by a margin of 12-14% especially when the attributes are highly skewed e.g. "country" and have very low assortativity values e.g. "subcategory". Low or negative assortativity implies easier availability to new content types and therefore faster reach of IXS to non-similar attribute values and therefore higher coverage.

From Equation (4), assuming that v has a unique content unseen by the IXS sampler. Therefore on all nodes ($\in \Delta v$), the Information surprise produced, $I_{\Delta v} \to \infty$, as $p_{\mathbb{S} \cup N(\mathbb{S})}(i) \to 0$ and $p_{\Delta v}(i) = 1/N$. Consequently, probability of nodes in the neighborhood of v gets sampled with more probability causing greater probability of v being included in the sample.

Table 6 shows that content-agnostic samplers are significantly better at preserving the network structure (network data) such as degree distribution, clustering coefficient and path length distribution. The bias of the content-aware samplers as illustrated in Section 3.2 makes them unsuitable for preserving the network structure. As expected UNI performs worst amongst all the samplers due to large number of disconnected components.

From Table 7, we observe a lack of any existing sampler that is efficient at preserving the content and network relationship. Therefore, it still remains an open problem to design efficient samplers that can preserve higher order content and network relationship such as global assortativity or local assortativity distribution.

In sum, UNI performs best for content distribution sampling, content-aware samplers are efficient at exploring new content values, content-agnostic samplers preserve the network structure and there is no statistically dominant strategy for preserving content-network relationship. Influenced by these findings, we believe that there is no single best sampling strategy. Departing from the conventional wisdom of

Table 5: The data coverage goal is to have maximum possible mean average coverage of content (1). IXS is shown as the greedy approach for unique content sampling. XS performs significantly better for attributes that are distributed across network community. content-aware samplers are statistically better than content-agnostic samplers. Abbrev: # *forward*: average emails forwarded by the user, *time*: average time between two emails

	Facebook		*Patent*		*Enron*	
Samplers	gender	locale	subcategory	in-citations	# forward	time
BFS	0.998*	0.292*	0.859*	0.892	0.269	0.065
RW	**1.000**	0.307*	0.853*	0.900	**0.312**	**0.068**
MHRW	**1.000**	0.295*	0.853*	0.902	0.239	0.067
FF	**1.000**	0.309*	0.870*	0.900	0.302	**0.068**
XS	**1.000**	0.483	0.830*	0.786*	0.254	0.063
UNI	**1.000**	**0.485**	0.799*	**0.913**	0.098	0.055
VNS	**1.000**	0.392	0.835*	0.901	-	-
IXS	**1.000**	0.355	0.858	0.901	-	-
ExP	-	-	0.868	0.905	0.049	0.038
I&M	**1.000**	0.357	0.861	0.907	-	-
pBM	**1.000**	0.312	0.840	0.889	-	-
pBX	0.993	0.348	**0.893**	0.778	-	-
Mean	0.999	0.358	0.852	0.881	0.218	0.061

Table 6: The network feature goal is to have least possible mean average KS statistic (0) of degree, clustering coefficient and pathlen for the preservation of network structure. Data agnostic samplers such as RW and MHRW perform best. Lack of any categorical attribute in Enron dataset makes some of the samplers in-feasible to implement in some of the cases. CC : clustering coefficient

	Facebook			*Patent*			*Enron*		
Samplers	Degree	CC	Path	Degree	CC	Path	Degree	CC	Path
BFS	0.403	0.295	0.781*	0.074*	0.075*	0.688*	0.342	0.295	0.679*
RW	**0.334**	0.257	0.636*	0.080*	0.080*	0.442*	0.391*	0.324*	0.423
MHRW	0.349	**0.209**	0.461	0.071*	0.072*	0.492*	0.315	0.308	**0.178**
FF	0.336	0.234	0.546	0.185*	0.188*	0.916*	0.388*	0.335*	0.387
XS	0.666*	0.522*	**0.100**	0.114*	0.040	0.554*	**0.151**	**0.122**	0.251
UNI	0.756*	0.623*	0.865*	0.863*	0.673*	0.988*	0.624*	0.518*	0.894*
VNS	0.533	0.349*	0.268	**0.036**	**0.032**	**0.175**	-	-	-
IXS	0.456	0.293	0.564	0.068	0.067	0.366	-	-	-
ExP	-	-	-	0.051	0.044	0.217	0.367	0.303	0.610
I&M	0.426	0.267	0.546	0.053	0.047	0.205	-	-	-
pBM	0.610	0.430	0.757	0.104	0.054	0.617	-	-	-
pBX	0.627	0.459	0.259	0.130	0.058	0.627	-	-	-
Mean	0.500	0.358	0.526	0.152	0.119	0.524	0.368	0.315	0.489

Table 7: The data-network relationship (assortativity) goal is to have least possible absolute difference in assortativity of attributes (0). By expectation of edges, Random edge sampling would be the best. There is no single dominant sampler strategy for this task. FB: Facebook, subcat: subcategory

	Facebook		*Patent*		*Enron*	
Samplers	gender	locale	subcategory	in-citations	# forward	time
BFS	**0.056**	0.167*	0.007*	0.082*	0.030	0.022
RW	0.066	0.188*	0.005	0.056	0.017	0.011
MHRW	**0.056**	0.131	0.005	0.040	0.018	0.021
FF	0.065	0.145	0.006	0.050	0.018	**0.010**
XS	0.091*	0.172*	0.012*	0.075*	**0.014**	0.016
UNI	0.074*	**0.086**	0.021*	**0.017**	0.057	0.045*
VNS	0.076	0.174*	0.005	0.043	-	-
IXS	0.066	0.139	0.005	0.061	-	-
ExP	-	-	**0.004**	0.067	0.068	0.015
I&M	0.065	0.157	**0.004**	0.080	-	-
pBM	0.080	0.185	0.007	0.117	-	-
pBX	0.076	0.121	0.012	0.070	-	-
Mean	0.070	0.151	0.008	0.063	0.032	0.020

designing the best (optimal) sampler for a given task, we emphasize on the need of better sampler choice dependent on the task at hand. In the next section, we focus on preserving content structure – clusters and classes in content.

5. LEARNING FROM DATA

In the last section, we saw no single strategy was best for preserving all the properties, but individual strategies typically outperformed others for specific tasks. In the first part of this section, we discuss the effect of sampling on cluster discovery and identification on real world and synthetic datasets. In the next part, we discuss the sampling effect on data classification.

Clustering is a common statistical tool for grouping similar objects into groups or clusters. The efficiency of a sampler lies in its ability to preserve the content structure. Given a group of n objects $x_1, x_2, x_3, ...x_i.., x_n$ and a distance measure d between any two data-points such that $d(x_i, x_j) \geq 0$, the clustering outputs object grouping S. The aim of clustering is therefore to preserve the object grouping S_k of the sampled nodes of size k with that of the original object grouping.

5.1 Experimental setup

Dataset description: We employ synthetic network generation model discussed in Section 2.2 to generate attributed network structure from several network generator as LFR, Watts-Strogatz and Barabasi model with varying data characteristics such as purity, skew and assortativity. Additionally, we assume that each object belongs to only one cluster and each node has two continuous and one discrete attribute values that are controlled by parameters of purity, skew and assortativity.

Evaluation: We use different clustering metrics for synthetic data, where the number of clusters is known, than real world datasets where the number of clusters is not known. We shall use Normalized Mutual Information (NMI) [19] for synthetic data and two measures for real world networks – Silhouette coefficient [19] and cluster coverage. *Cluster coverage* is defined over the discrete attributes in Facebook and Enron, as the ratio of sampled nodes with unique attribute combinations to all possible attribute combinations in the underlying network. Similar to the last section, we take the mean average of 100 runs of simulations over k =1, 2, 3, till 10% sample size.

5.2 Experimental results

Now we present the clustering results on synthetic data where we have systematically varied network structure as well as properties of the attributes. We shall conclude with clustering results on real world network by drawing an analogy between the synthetic network and real-world network.

Network effect: Figure 3 depicts the sampling performance over different sampling strategies on three synthetic networks: Barabasi, Watts-Strogatz ($p = 0.1$, high clustering coefficient) and LFR($\mu = 0.1$, high clustering with power law degree distribution). The relative performance as seen in the figure for very different strategies remains very similar. Simulations over a series of synthetic networks obeying real world network properties and real world networks, lead us to the conclusion that network structure has very little effect on the relative performance of samplers.

Data dependence Table 8 shows the clustering performance of different samplers on a real world-resembling network under different data characterizations. In all cases, content-aware samplers perform the best at clustering. More importantly they are statistically better than the content-agnostic samples at 95% confidence level. In the best scenario, content-aware sampler outperforms content-agnostic samplers by more than 20% (s_m, a_h, p_l).

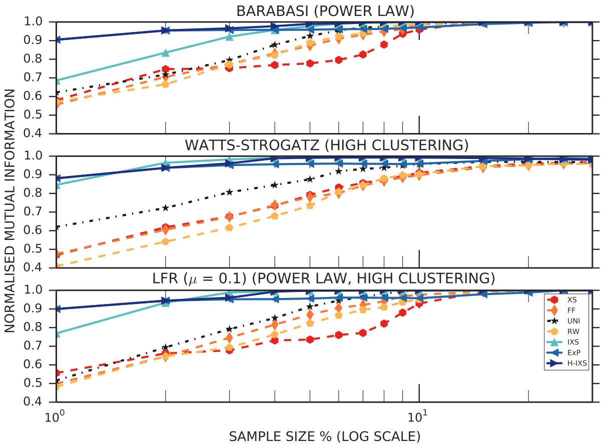


Figure 3: Normalized mutual information for three networks exhibiting real world characteristics display similar sampling performance. The data parameter value used are high skew (s_h), high assortativity (a_h) and high purity (p_h).

All samplers perform better with increased *purity*. This is because increased purity implies that the clusters are well separated. So it is unsurprising that when purity is high, skew and assortativity become less relevant. However, we do see samplers other than (IXS, H-IXS and ExP) slowly degrade their performance with increasing skew and increased assortativity. All samplers have their best performance at low skew (all attribute values are equally probable), high assortativity and high purity. The worst performance is in the case of low assortativity, high skew and low purity.

Observing the three blocks of columns sized-four from left to right, we clearly see that skewness makes it harder for the samplers to preserve the clustering. It is unclear as to why

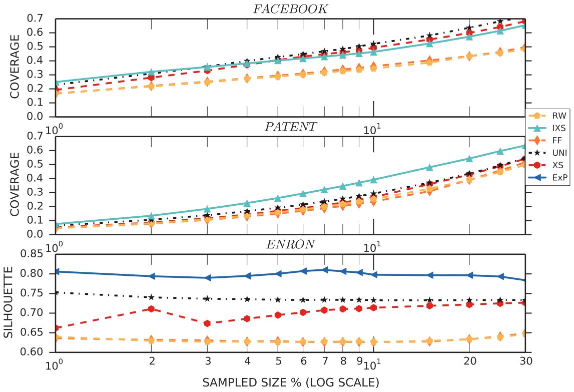


Figure 4: Clustering performance on three real world datasets – Facebook, Patent and Enron shows content agnostic samplers being outperformed by content-aware ones.

the performance slightly improves at mid-skew levels which was expected to be slightly worse off. This may be due to an interplay of several data characteristics which is not fully understood. In general, as skewness in the size of data cluster increases, it makes it increasingly difficult to identify smaller sized clusters that are overwhelmed by larger sized cluster.

We see from the results in 8 that high values of assortativity causes performance degradation at mid-skew and high-skew levels by acting as a bottleneck for link-trace samplers. Consequently, we can observe that at mid-skew and high-skew levels and at high-purity levels, greater randomness ($a \rightarrow 0$) leads to greater correct (p_h) cluster information and therefore better performance. Conversely at low-purity levels, the NMI performance degrades owing to the fact that now greater information means means more noisy information. Therefore, assortativity controls the accessibility to different data clusters as the different network samplers spread across the network.

Table 9: Network joint attribute property. JC: joint cardinality of the attributes is the prodcut of their individual cardinality

Networks	Attributes	JC	Skew	Assortativity
Facebook	gender, locale, education;type	35	0.47	0.11
Patent	cat, subcategory, assignee type, country	9472	0.18	0.03

Figure 4 shows the clustering performance on real world networks. As described in Section 5.1, coverage of data-cluster is used for evaluation on Facebook and Patent network whereas silhouette coefficient for Enron network. ExP and IXS work best for Facebook network where data clusters are partially distributed across network as community alongwith XS and UNI, that outperform traditional link by 20%. IXS is the best sampler for Patent by 12% margin. The table 8 can be used to predict the performance of samplers by comparing our synthetic network model: (s_m, a_l, p_h) to Facebook and (s_l, a_l, p_h) to Patent. ExS is the best sampler for Enron outperforming by 5-6% over UNI and 18% over RW. In absence of any known k (number of data clusters) for Enron dataset, we probe into the effect of k at a fixed sample budget of 10% and variable k' performance of all samplers(5) suggesting that the relative performance of samplers remains unchanged by varying k. At lower k values, outlier sampling methodology seems to cause poor performance in ExP probably because it picks a lot of outliers and causes confusion among the very few cluster centers.

In this section, we we found that content-aware samplers were superior to content-agnostic ones in a statistically significant sense. In the next section, we briefly describe the impact of sampling on another data mining task—classification.

6. CLASSIFICATION

Given a set of feature n vectors X_i and their target class Y_i, the objective of classification is to learn a classifier function f that minimizes the loss in prediction of target class on unseen data points X_u. i.e. $\mathbb{R} = \mathbb{E}(l(Y_u - f(X_u)))$ the expected loss (l) or risk ($\mathbb{R}$) is minimized. Classification differs from clustering in the last section based to the fact that it uses training examples to predict future classes, whereas clustering has no prior information about the clusters and is completely unsupervised.

Dataset description: We re-use the synthetic attributed network generator from Section 2.2 along with the real world dataset in this section. To avoid unwanted attribute value or type biasness, we construct the feature vectors from the continuous and discrete attribute values using well known z-score standardization and one-hot encoding respectively [20].

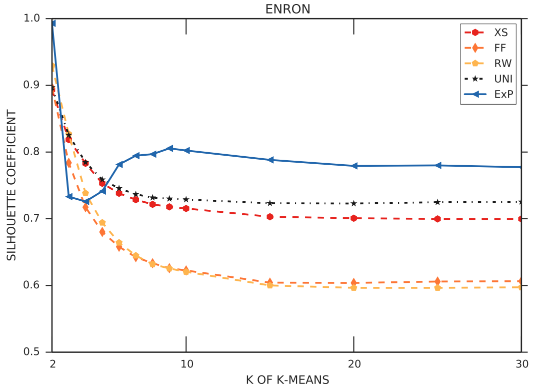


Figure 5: Clustering performance in Enron network by varying k of k-means reveal the dependence of sampler performance on number of clusters.

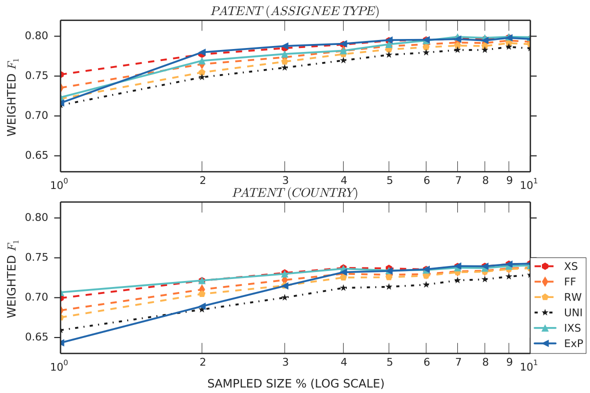


Figure 6: Classification performance for prediction of attribute “assignee type” and “country” in Patent dataset shows the performance of content-aware samplers.

Classification Objective: The classification performance of a sampler is its ability to sample discriminative data samples that helps learn the better classifier function. There are two scenarios that may happen depending on whether the target class is known during sampling or not. Irrespective of the two scenarios, we execute our content-aware samplers on all attributes (target class and/or features). When the target class is known during sampling, a balancing of the target class helps in classifier performance than when target class is unknown. However, the relative sampler performances remain unchanged as illustrated through Table 10. We therefore show the results of classification on real world networks for attribute prediction when the attribute or target class is known during sampling.

We use the *weighted* F_1 score [19] for the classification accuracy on the real world attribute prediction for Facebook and Patent because the attributes are discrete and skewed and, thus *weighted* F_1 helps in balancing each class's performance making it suitable for even skewed class prediction. We use the R^2 coefficient of determination for evaluating the performance of regression on the continuous attribute in Enron.

We conducted similar synthetic experiments as shown in the last section for predicting the "hidden" cluster `id` of the nodes. We observed similar results in the relative performance of the samplers on classification and clustering since target property to be discovered remains same. We however found that content-aware sampling on both attributes and target class increases the classification results significantly.

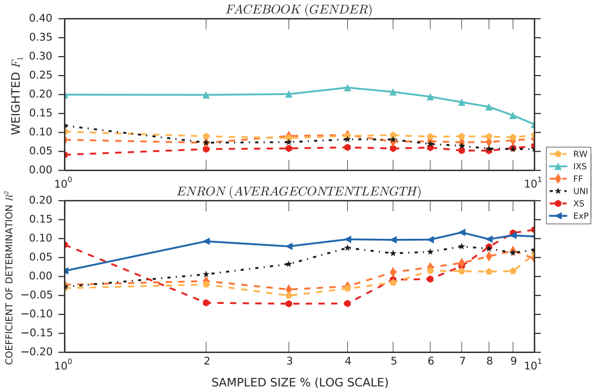


Figure 7: Classification performance on two real world datasets. The classification performance on Facebook and Enron shows content agnostic samplers being outperformed by content-aware samplers only when the attribute are target attribute is correlated to at least one of the feature attributes.

From Figure 7, we see that the classification result on prediction of "gender" in the Facebook network from the remaining attributes "locale" and "education type" under which condition IXS performs better than other competing samplers by 10-12% of weighted F_1 score. We also observe the SVR prediction of "average content length" attribute of an Enron user from the other attributes in Enron dataset. From the inverted correlation matrix of attributes of Enron, we find no significant correlation among attributes, and possibly this attribute independence leads to the near equal performance of all samplers on SVR [19]. From Figure 6, we similarly the results show the prediction results from predicting "country" and "assignee type" attributes from the remaining attributes of patent in Patent network.

7. RELATED WORK

Not surprisingly, data inference from data gathered by network sampling arises in many diverse areas. We study the prior works related to our work in three different areas: network sampling, data sampling and joint network-data sampling

"Representative subgraph" sampling closes resembles to our methodology of sampling attributed network. Representative subgraph sampling aims to construct a sampled subgraph that has network structure very similar to that of the original network. Forest Fire Leskovec and Faloutsos [4] preserved several key network structure characteristics. Hubler et al. [14] showed via Metropolis algorithm that prior knowledge of the network can help in obtaining better representative samples. The objective of our work is preservation of content properties and not just the network structure.

Another line of research on network sampling focuses on understanding the biases of existing samplers and ways to obtain uniform samples. Kurant et al. [15] quantified the degree bias for several network samplers and proposed new ways to correct them. Costenbader and Valente [21] did thorough analysis of the effect of noise (sample) on network centrality estimation. Gjoka et al. Gjoka et al. [13] implemented the proposed uniform link-trace samplers on very massive Facebook network to validate the results. Chiericetti et al. [22] proposed an efficient random walk sampling strategy for sampling according to a prescribed distribution not just "uniform" sampling. Maiya and Berger-Wolf [6] exploited the bias instead of correcting it to design expansion based samplers. Similar to Maiya's work, we exploit the bias of entropy based samplers to balance the attributes in the sample, yielding improved classification and clustering performance.

There has been a plethora of research on sampling data from an unknown population distribution. Our objective resembles with these surveys that try to estimate the underlying data characteristics. However most the well known samplers such as Poisson sampling, stratified sampling, etc. Patton [23] requires random access to the nodes in dataset and therefore fail to work upon network structure. The idea of surprise based data sampling is however not new. It has been used in the fields of graph visualization, information retrieval, active learning, etc. For example, in classical database search, Sarwagi Sarawagi [24] used the Maximum Entropy principle to model a user's knowledge and aid the user in exploring OLAP data cubes. In graph visualization work Pienta et al. [25], the authors chose to highlight the neighbors that are most surprising in information. Our work borrows the idea of surprise defined in terms of entropy and stratified sampling principles to design better attributed network samplers.

Sociological and statistical studies on social networks such as friendship recommendation, link prediction, attribute inference, type distribution, etc. implicitly rely upon both content and network. However very few research has been done to understand the effect of sampling on joint network and content characteristics. Li et al. Li and Yeh [26] studied five different sampling strategies for node-type and link-type distribution preservation. They noted that sample size of 15% from RDS, the best sampling strategy, can preserve "location type" distribution in Twitter network very well. Yang et al. Yang et al. [27] proposed a semantic sampling strategy, Relational Profile sampling, that preserves the semantic relationship types in a heterogeneous networks. Park et al. Park and Moon [28] remarked about the inefficiency of the existing network samplers in estimating node attributes. Although seemingly similar to these work, our work is different in its objective. To the best of our knowledge, we are the first to propose network samplers for data mining purposes like clustering and classification.

8. LIMITATIONS

In this section we discuss several limitations of this work. First, much of the analysis assumes that we have no missing values; while the algorithms would work in the case of missing values, it would useful to introduce a noise model to formally estimate error in surprise when confronted with missing values. Second, the time and space complexity of IXS is greater than MHRW and RW. The incremental update complexity is $O(\mu \log |\mathbb{S}| + \mu^2)$, where μ is the mean degree, while it is $O(1)$ for RW or MHRW. Some of this can be mitigated by appropriate data structures. For example, we commonly assume that a node has access to the `id`'s of its neighbors, but not the attributes of its neighboring nodes; this can be easily rectified, reducing the incremental time complexity. Third our model of link-trace sampling is limited: many social networks allow us to make queries on network data returning network nodes that satisfy the query. It would be interesting to expand the sampling paradigm to incorporate a more rich query model. Finally, it would nice to have a theoretical argument why we have empirically observed that no sampler performs best in all situations.

9. CONCLUSION

In this paper, we have presented new data aware sampling methodologies for attributed networks. The problem is important because data mining tasks such as clustering or classification are commonplace on the nodal attributes of real-world networks. A key challenge is that these large networks are often sampled with BFS or RW, which were never designed to preserve content characteristics. In the first of its kind study, we show that these samplers are suboptimal for standard data mining tasks. We proposed several samplers based on the idea of information expansion. We have excellent results with information based sampling outperforming the baselines for all tasks in a statistically significant manner, with best-case performance improvements of 25%.

References

[1] Ibrahim Sorkhoh, Maytham Safar, and Khaled Mahdi. "Classification of social networks". In: *IADIS International Conference WWW/Internet.* 2008.

[2] Jaewon Yang, Julian McAuley, and Jure Leskovec. "Community detection in networks with node attributes". In: *Data Mining (ICDM), 2013 IEEE 13th international conference on.* IEEE. 2013, pp. 1151–1156.

[3] Mark S Handcock, Adrian E Raftery, and Jeremy M Tantrum. "Model-based clustering for social networks". In: *Journal of the Royal Statistical Society: Series A (Statistics in Society)* 170.2 (2007), pp. 301–354.

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

[5] Christian Hubler, Hans-Peter Kriegel, Karsten Borgwardt, and Zoubin Ghahramani. "Metropolis algorithms for representative subgraph sampling". In: *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on.* IEEE. 2008, pp. 283–292.

[6] Arun S Maiya and Tanya Y Berger-Wolf. "Benefits of bias: Towards better characterization of network sampling". In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM. 2011, pp. 105–113.

[7] Julian J McAuley and Jure Leskovec. "Learning to Discover Social Circles in Ego Networks." In: *NIPS.* Vol. 2012. 2012, pp. 548–56.

[8] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. "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.* ACM. 2005, pp. 177–187.

[9] Jure Leskovec, Kevin J Lang, Anirban Dasgupta, and Michael W Mahoney. "Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters". In: *Internet Mathematics* 6.1 (2009), pp. 29–123.

[10] Andrea Lancichinetti and Santo Fortunato. "Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities". In: *Physical Review E* 80.1 (2009), p. 016118.

[11] Konstantinos Pelechrinis. "Matching patterns in networks with multi-dimensional attributes: a machine learning approach". In: *Social Network Analysis and Mining* 4.1 (2014), pp. 1–11.

[12] Mark EJ Newman. "Mixing patterns in networks". In: *Physical Review E* 67.2 (2003), p. 026126.

[13] M. Gjoka, M. Kurant, C.T. Butts, and A. Markopoulou. "Walking in Facebook: A Case Study of Unbiased Sampling of OSNs". In: *INFOCOM, 2010 Proceedings IEEE.* 2010, pp. 1–9.

[14] C. Hubler, H.-P. Kriegel, K. Borgwardt, and Z. Ghahramani. "Metropolis Algorithms for Representative Subgraph Sampling". In: *Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on.* 2008, pp. 283–292.

[15] Maciej Kurant, Athina Markopoulou, and Patrick Thiran. "On the bias of bfs (breadth first search)". In: *Teletraffic Congress (ITC), 2010 22nd International.* IEEE. 2010, pp. 1–8.

[16] Morris H Hansen and William N Hurwitz. "On the theory of sampling from finite populations". In: *The Annals of Mathematical Statistics* 14.4 (1943), pp. 333–362.

[17] Miller McPherson, Lynn Smith-Lovin, and James M. Cook. "Birds of a Feather: Homophily in Social Networks". In: *Annual Review of Sociology* 27 (2001), pp. 415–444.

[18] Suhansanu Kumar and Hari Sundaram. "Task Driven Sampling of Attributed Networks". In: *ArXiv e-prints* (2016).

[19] Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques.* Elsevier, 2011.

[20] Joseph E Beck and Beverly Park Woolf. "High-level student modeling with machine learning". In: *International Conference on Intelligent Tutoring Systems.* Springer. 2000, pp. 584–593.

[21] Elizabeth Costenbader and Thomas W Valente. "The stability of centrality measures when networks are sampled". In: *Social networks* 25.4 (2003), pp. 283–307.

[22] Flavio Chiericetti, Anirban Dasgupta, Ravi Kumar, Silvio Lattanzi, and Tamás Sarlós. "On sampling nodes in a network". In: *Proceedings of the 25th International Conference on World Wide Web.* International World Wide Web Conferences Steering Committee. 2016, pp. 471–481.

[23] Michael Quinn Patton. *Qualitative research.* Wiley Online Library, 2005.

[24] Sunita Sarawagi. “User-Adaptive Exploration of Multidimensional Data.” In: *VLDB.* 2000, pp. 307–316.

[25] Robert Pienta, Zhiyuan Lin, Minsuk Kahng, Jilles Vreeken, Partha P Talukdar, James Abello, Ganesh Parameswaran, and Duen Horng Polo Chau. “AdaptiveNav: Discovering Locally Interesting and Surprising Nodes in Large Graphs”. In: ().

[26] Jhao-Yin Li and Mi-Yen Yeh. “On sampling type distribution from heterogeneous social networks”. In: *Advances in Knowledge Discovery and Data Mining.* Springer, 2011, pp. 111–122.

[27] Cheng-Lun Yang, Perng-Hwa Kung, Chun-An Chen, and Shou-De Lin. “Semantically sampling in heterogeneous social networks”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* International World Wide Web Conferences Steering Committee. 2013, pp. 181–182.

[28] Hosung Park and Sue Moon. “Sampling bias in user attribute estimation of OSNs”. In: *Proceedings of the 22nd international conference on World Wide Web companion.* International World Wide Web Conferences Steering Committee. 2013, pp. 183–184.

Table 8: The clustering preservation goal is to have highest possible NMI score (1). The table shows the different samplers' performances on a real world resembling network (LFR $\mu = 0.1$). It is evident from the table that the parameters of skew ($s_l = 0,\ s_m = 0.22,\ s_h = 0.52$), purity ($p_l = 0.2,\ p_h = 10$) and assortativity ($a_l = 0,\ a_h = 1$) have a significant impact on the clustering performance (NMI). * indicates that the best performing content-aware sampler statistically significantly outperforms the star-ed samplers (content-agnostic) by a confidence of 95%.

	s_l				s_m				s_h			
	a_l		a_h		a_l		a_h		a_l		a_h	
samplers	p_l	p_h	p_l	p_h	p_l	p_h	p_l	p_h	p_l	p_h	p_l	p_h
BFS	0.386*	0.979	0.371*	0.99	0.397*	0.967	0.396*	0.951*	0.274*	0.901	0.303*	0.836*
RW	0.379*	0.977	0.379*	0.993	0.412*	0.975	0.393*	0.950*	0.286*	0.925	0.305*	0.837*
MHRW	0.382*	0.977	0.388*	0.992	0.405*	0.968	0.393*	0.948*	0.274*	0.890*	0.307*	0.844*
FF	0.372*	0.977	0.378*	0.992	0.408*	0.966	0.393*	0.953*	0.275*	0.921	0.305*	0.845*
XS	0.383*	0.973	0.366*	0.991	0.419*	0.984	0.380*	0.953*	0.280*	0.964	0.301*	0.823*
UNI	0.388*	0.981	0.378*	**0.996**	0.409*	0.965	0.390*	0.961*	0.279*	0.891*	0.316*	0.857*
VNS	0.385*	0.98	0.373*	**0.996**	0.404*	0.953*	0.403*	0.961*	0.280*	0.837*	0.295*	0.848*
IXS	0.385	**0.985**	0.383	**0.996**	0.382	0.982	0.403	0.973	0.275	0.933	0.323	0.962
ExP	**0.566**	0.955	**0.558**	0.961	**0.594**	0.951	**0.581**	0.974	**0.447**	0.934	**0.486**	0.931
H-IXS	0.469	0.98	0.458	0.99	0.495	**0.985**	0.493	**0.985**	0.402	**0.975**	0.431	**0.973**
B&M	0.382	0.98	0.38	0.994	0.404	0.973	0.408	0.965	0.267	0.914	0.313	0.918
pBM	0.396	0.983	0.375	0.982	0.405	0.956	0.393	0.928	0.283	0.84	0.301	0.779
pBX	0.387	0.977	0.366	0.993	0.42	0.983	0.379	0.956	0.274	0.965	0.3	0.827
Mean	0.405	0.977	0.396	0.99	0.427	0.97	0.416	0.958	0.3	0.915	0.33	0.868

Table 10: The classification preservation goal is to have highest possible weighted F_1 score (1). The table shows the different samplers' performances in predicting the hidden cluster-`id` on a real world resembling network (LFR $\mu = 0.1$). It is evident from the table that the parameters of skew ($s_l = 0,\ s_m = 0.22,\ s_h = 0.52$), purity ($p_l = 0.2,\ p_h = 10$) and assortativity ($a_l = 0,\ a_h = 1$) have a significant impact on the classification performance.

	s_l				s_m				s_h			
	a_l		a_h		a_l		a_h		a_l		a_h	
samplers	p_l	p_h	p_l	p_h	p_l	p_h	p_l	p_h	p_l	p_h	p_l	p_h
BFS	0.296	0.929	0.297	0.876	0.39	0.918	0.451	0.89	0.717	0.913	0.712	0.884
RW	0.294	0.907	0.303	0.896	0.397	0.915	0.456	0.894	0.715	0.929	0.714	0.881
MHRW	0.282	0.915	0.306	0.897	0.385	0.916	0.455	0.902	0.715	0.905	0.714	0.882
FF	0.291	0.918	0.302	0.897	0.395	0.911	0.46	0.903	0.715	0.925	0.712	0.877
XS	0.314	0.932	0.288	0.88	0.407	0.927	0.434	0.898	0.718	0.946	0.716	0.878
UNI	0.292	0.917	0.311	0.9	0.394	0.905	0.454	0.904	0.712	0.908	0.711	0.889
VNS	0.3	0.911	0.312	0.916	0.387	0.918	0.452	0.918	0.715	0.908	0.717	0.908
IXS	**0.367**	**0.972**	**0.391**	**0.960**	**0.449**	**0.964**	**0.514**	**0.951**	0.728	**0.961**	0.723	0.939
ExP	0.308	0.755	0.335	0.765	0.403	0.851	0.481	0.825	**0.754**	0.952	**0.749**	0.932
H-IXS	0.266	0.864	0.278	0.844	0.357	0.891	0.427	0.891	0.72	0.946	0.711	0.935
B&M	0.281	0.917	0.303	0.892	0.395	0.948	0.445	0.94	0.72	0.952	0.713	**0.945**
pBM	0.296	0.917	0.299	0.883	0.399	0.9	0.453	0.874	0.713	0.886	0.71	0.868
pBX	0.295	0.914	0.308	0.9	0.405	0.926	0.437	0.897	0.715	0.948	0.716	0.882
Mean	0.299	0.905	0.31	0.885	0.397	0.915	0.455	0.899	0.72	0.929	0.717	0.900