FAIRNESS-AWARE SELECTIVE SAMPLING ON ATTRIBUTED GRAPHS

Oyku Deniz Kose, Yanning Shen*

Department of Electrical Engineering and Computer Science, University of California Irvine, USA

ABSTRACT

Selective sampling is an online learning framework where the learner tries to detect the data samples whose labels can boost the performance maximally, and only the labels of chosen data samples are queried. While the design of selective sampling algorithms is extensively studied for independent data samples, the area is rather under-explored in the context of graphs. Furthermore, the limited number of existing graph-based approaches do not take into consideration the nodal attributes that are available in attributed graphs. In this study, existing online learning and selective sampling algorithms are modified to be used with graphs that have nodal features. Additionally, the bias in the results of original algorithms is investigated, and a novel bias reduction strategy is proposed that can be embedded into the dimensionality reduction step without incurring a significant complexity cost. Experiments for node classification are carried out on real social networks to showcase the advantages of incorporating nodal features, as well as the fairness-enhancement framework.

1. INTRODUCTION

Graphs are widely used to represent complex underlying relations in several real-world systems such as social networks [1], power-grids [2], and gene networks [3]. Their use in a wide range of applications has made the studies on graphs of particular interest, yet processing graphs presents unique challenges. For example, the prevalent assumption of independent and identically distributed (i.i.d.) data cannot be made for graph-structured data, which shows the need for the consideration of graph structure in the design of algorithms. This motivates the study of semi-supervised learning over graphs [4].

Node classification on graphs enables a number of real-world applications, such as fraud detection [5], and disease prediction [6]. Learning algorithms for node classification can be broadly classified into two, online and offline algorithms. In the offline setting, the whole data for the nodes is available to the learner in the beginning, and the labels of the nodes can be queried any time [7, 8]. This learning scheme requires considerable amount of memory, as the data should be accessible throughout the learning process. In the online learning, data samples are received by the learner in a sequential order, and after the model is updated according to the current sample, that data sample can be discarded [9, 10]. This sequential manner makes online learning a better alternative to study on the big data.

While online learning algorithms provide a more efficient framework in terms of memory complexity, they assume that all the labels for nodes are available. This assumption can be restrictive for certain applications, as the labeled data is scarce and expensive to produce. For this reason, active learning methods have been studied extensively on both i.i.d. [11, 12, 13] and graph data [14, 15, 16]. Active learning aims to reduce the need for labeled data by determining data samples whose labels can improve the performance maximally, and querying only the labels of chosen data samples. Selective sampling can be viewed as the unification of online learning and active sampling [17, 18, 19], in which the data samples are obtained in a sequential order by the learner, and the learner tries to achieve a trade-off between the performance and the number of queried labels.

Existing works on selective sampling and online learning over graphs for the node classification task do not consider the existence of any nodal attributes, and the algorithms are designed to cope with information coming from the graph structure only [20, 21, 22]. Therefore, they may not fully utilize available information from nodal features that might be useful for the learning task, in cases where the graph is attributed.

Meanwhile, it has been shown that machine learning algorithms can propagate bias within the data towards under-represented groups [23, 24, 25]. Such bias may lead to unfair decision making in areas where the algorithms are applied, e.g., criminal justice. Furthermore, it has been demonstrated that the utilization of graph structure magnifies the existing bias, as the nodes in a graph tend to form relations based on the sensitive attributes (e.g., gender, race) [1, 26, 27]. Hence, a fairness-enhancement strategy is needed to mitigate the bias. To this end, the main contributions of this study are as follows:

1. We improve existing online learning and selective sampling algorithms to exploit nodal information in attributed graphs.
2. We develop a novel fairness enhancement scheme to mitigate the bias towards certain sensitive groups without incurring a significant additional complexity. The proposed scheme is embedded into the dimensionality reduction step, thus it can be flexibly utilized in any algorithm that employs such a step.
3. The experiments on real social networks show that the utilization of nodal attributes in a correct way can indeed improve the performance, albeit at the cost of amplified bias. Our results demonstrate that this bias can be effectively mitigated by the introduced fairness enhancement approach. Overall, it is observed that using the two introduced approaches jointly provides both lower error rates and better fairness metrics.

2. METHODOLOGY

2.1. Preliminaries

Throughout this paper, the lower case letters refer to scalars, while the lower case and upper case bold letters denote vectors and matrices, respectively. For a matrix $\mathbf{\Delta} \in \mathbb{R}^{N \times N}$, $\Delta_{i,j}$ denotes the entry of $\mathbf{\Delta}$ in row i and column j, and $\mathbf{\Delta}^{\dagger}$ is the pseudo-inverse of $\mathbf{\Delta}$. Input graph can be represented with $\mathcal{G} := (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} := \{v_1, v_2, \dots, v_N\}$ denotes the node set, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ refers to the edge set. Matrices $\mathbf{X} \in \mathbb{R}^{N \times F}$ and $\mathbf{A} \in \{0,1\}^{N \times N}$ represent the nodal feature and adjacency matrices, respectively, where $A_{ij} = 1$

Work in this paper was partially supported by Microsoft Academic Grant Award for AI Research.

if and only if $(v_i, v_j) \in \mathcal{E}$. Furthermore, diagonal graph degree matrix is denoted with $\mathbf{D} \in \mathbb{R}^{N \times N}$ where each diagonal entry is the degree of the corresponding node, and $\mathbf{L} \in \mathbb{R}^{N \times N}$ represents normalized graph Laplacian matrix which equals to $\mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$. Finally, sensitive attributes and labels of the nodes are represented by $\mathbf{s} \in \{0,1\}^{N \times 1}$, and $\mathbf{y} \in \mathbb{R}^{N \times 1}$, respectively. In this work, the existence of a single, binary sensitive attribute is considered.

2.2. Problem Formulation

Online and selective sampling studies on graphs make use of the information coming from the graph structure through the concept of learning with local and global consistency (LLGC) [8]. Specifically, by introducing a Laplacian regularizer [28], the learning algorithm tends to assign similar labels to nodes that are connected . The graph Laplacian regularizer can be written as follows

$$\frac{1}{2}\sum_{i,j=1}^{n}(f_i - f_j)^2\mathbf{A}_{ij} = \mathbf{f}^\top\mathbf{L}\mathbf{f}, \tag{1}$$

where f_i denotes the estimated label for node i. Together with the fitting error, the problem of learning with LLGC can be written as

$$\min_{\mathbf{f}} \frac{1}{2}\|\mathbf{f} - \mathbf{y}\|^2 + \frac{\mu}{2}\mathbf{f}^\top\mathbf{L}\mathbf{f} \tag{2}$$

which can then be reformulated as

$$\min_{\mathbf{w}} \frac{1}{2}\left\|\mathbf{M}^\top\mathbf{w} - \mathbf{y}\right\|^2 + \frac{\mu}{2}\|\mathbf{w}\|^2, \tag{3}$$

where $\mathbf{L}^\dagger = \mathbf{M}^\top\mathbf{M}$, $\mathbf{w} = \mathbf{M}\boldsymbol{\alpha}$, and $\mathbf{f} = \mathbf{L}^\dagger\boldsymbol{\alpha}$. Note that the equality $\mathbf{L}^\dagger = \mathbf{M}^\top\mathbf{M}$ can be written, as $\mathbf{L}^\dagger$ is positive semi-definite (PSD). Here, $\mathbf{M} \in \mathbb{R}^{N \times N}$ is the data matrix whose columns are data vectors and $\mathbf{w} \in \mathbb{R}^N$ is the linear classifier used for node classification. In the sequel, we will focus on the generation of data matrix $\mathbf{M}$ using both graph structure and nodal features.

2.3. Fusion of Nodal Features

While existing algorithms are quite efficient in exploiting information of the graph structure through the Laplacian regularizer, they do not consider the nodal features which may carry essential information for the learning task at hand. In this study, we will first introduce a feature fusion framework to incorporate the nodal features together with the graph structure information. Note that the fusion of the graph topology information and the nodal attributes is not straightforward. Directly concatenating data vectors $\mathbf{m}_i$ and nodal features $\mathbf{x}_i$ may not provide the optimal performance, and such an approach also prevents the direct extension of the theoretical guarantees presented in [21] and [22]. To this end, this work aims to design a fusion framework to blend the information coming from two sources.

Specifically, instead of solely relying on graph Laplacian $\mathbf{L}$ in the generation of data matrix $\mathbf{M}$ where $\mathbf{L}^\dagger = \mathbf{M}^\top\mathbf{M}$, information coming from both the graph structure $\mathbf{L}$ and the nodal features $\mathbf{L_x}$ are utilized such that

$$(\mathbf{L} + \beta\mathbf{L}_x)^\dagger = \hat{\mathbf{M}}^\top\hat{\mathbf{M}} \tag{4}$$

where $\hat{\mathbf{M}} \in \mathbb{R}^{N \times N}$ is the modified data matrix, $\mathbf{L}_x$ is a Laplacian like matrix generated for the nodal features, and β is a hyperparameter. Note that $\mathbf{L}_x$ should also be a PSD matrix to assure a decomposition for the creation of the new data matrix $\hat{\mathbf{M}}$. This modification basically changes the graph regularizer term in the initial problem formulation, the new regularizer assures a smoothness over labels considering both the graph structure and the nodal attributes. The resulting problem becomes

$$\min_{\hat{\mathbf{f}}} \frac{1}{2}\|\hat{\mathbf{f}} - \mathbf{y}\|^2 + \frac{\mu}{2}\hat{\mathbf{f}}^\top(\mathbf{L} + \beta\mathbf{L}_x)\hat{\mathbf{f}}. \tag{5}$$

Considering (4) and letting $\hat{\mathbf{w}} = \hat{\mathbf{M}}\boldsymbol{\alpha}$, and $\hat{\mathbf{f}} = (\mathbf{L} + \beta\mathbf{L}_x)^\dagger\boldsymbol{\alpha}$, (5) can be re-written as follows

$$\min_{\hat{\mathbf{w}}} \frac{1}{2}\left\|\hat{\mathbf{M}}^\top\hat{\mathbf{w}} - \mathbf{y}\right\|^2 + \frac{\mu}{2}\|\hat{\mathbf{w}}\|^2 \tag{6}$$

which is a ridge regression problem. The corresponding online formulation can then be written as follows

$$\hat{\mathbf{w}}_{t+1} = \arg\min_{\hat{\mathbf{w}}} \frac{1}{2}\sum_{i=1}^{t}\left(\hat{\mathbf{m}}_i^\top\hat{\mathbf{w}} - y_i\right)^2 + \frac{\mu}{2}\|\hat{\mathbf{w}}\|^2, \tag{7}$$

where $\hat{\mathbf{m}}_t$ is the modified input vector and y_t denotes its label at time step t. The optimal solution of this problem is [29]

$$\hat{\mathbf{w}}_{t+1} = \left(\sum_{i=1}^{t}\hat{\mathbf{m}}_i\hat{\mathbf{m}}_i^\top + \mu\mathbf{I}\right)^{-1}\sum_{i=1}^{t}\hat{\mathbf{m}}_i y_i. \tag{8}$$

Note that the recursive updates on the classifier $\hat{\mathbf{w}}$ in (8) are modified based on the selective sampling algorithms utilized as baselines. In selective sampling, labels are inquired only when the confidence towards a prediction is lower than a certain threshold [21, 22]. Therefore, this work presents a data matrix generation step where the information of both nodal features and graph topology is used, and the created matrix can be employed in any graph-based algorithm built upon the problem formulation presented in Subsection 2.2.

In order to generate the $\mathbf{L}_x$ matrix, a cosine similarity-based similarity matrix for the standardized (zero mean, unit variance) nodal features $\tilde{\mathbf{X}}$ is obtained with the (i, j) th entry;

$$K_{i,j} = \frac{\mathrm{Sim}_{\cos}(\tilde{\mathbf{x}}_\mathbf{i}, \tilde{\mathbf{x}}_\mathbf{j}) + 1}{2}, \tag{9}$$

where $\mathrm{Sim}_{\cos}(\boldsymbol{a}, \boldsymbol{b}) := \frac{\boldsymbol{a}^\top\boldsymbol{b}}{\|\boldsymbol{a}\|\|\boldsymbol{b}\|}$ denotes the cosine similarity between vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, $\tilde{\mathbf{x}}_i$ and $\tilde{\mathbf{x}}_j$ are the standardized nodal attributes of nodes i and j, respectively. The entries of this similarity matrix $\mathbf{K}$ lie in the interval $[0, 1]$, and $\mathbf{K}$ is a PSD matrix that signifies the similarity of nodes in terms of feature vectors. Therefore, $\mathbf{K}$ resembles graph adjacency matrix and a normalized Laplacian-like matrix $\mathbf{L}_x$ can be generated based on $\mathbf{K}$ such that $\mathbf{L}_x = \mathbf{I} - \mathbf{K}_D^{-\frac{1}{2}}\mathbf{K}\mathbf{K}_D^{-\frac{1}{2}}$ where $\mathbf{K}_D$ is a diagonal matrix with diagonal entries $d_{i=}\sum_{j=1}^{N}K_{i,j}$, for $i = \{1, \dots, N\}$. Note that $\mathbf{L}_x$ matrix is PSD, which can be shown with the same way that is used to illustrate the positive semi-definiteness of the Laplacian matrix. Furthermore, while linearly combining $\mathbf{L}$ and $\mathbf{L}_x$, these matrices need to be normalized using $\mathbf{D}$ and $\mathbf{K}_D$ before fusion to avoid loss of information from the graph topology or nodal features.

2.4. Fairness Enhancement in Dimensionality Reduction

Data matrix generation based on (4) leads to N dimensional data vectors, with which the training of linear classifier $\hat{\mathbf{w}}$ has complexity $\mathcal{O}(N^2)$. Such a complexity can be restrictive while working on large graphs (i.e. N is large). Therefore, a dimensionality reduction step is required to generate $\hat{\mathbf{M}}^d \in \mathbb{R}^{d \times N}$ that consists of d dimensional data vectors $\hat{\mathbf{m}}_i^d \in \mathbb{R}^d, i = \{1, \dots N\}$ to make the learning algorithms computationally cheaper. Since $\hat{\mathbf{M}}^\top\hat{\mathbf{M}} = (\mathbf{L} + \beta\mathbf{L}_x)^\dagger$,

Algorithm 1: Fusion of Nodal Features and Graph Structure

```
Data: $\mathbf{L}, \mathbf{X}$ , $\beta$
Result: $\hat{\mathbf{M}}$
1. Standardize $\mathbf{X}$ to create $\tilde{\mathbf{X}}$
2. Generate similarity matrix $\mathbf{K}$ for $\tilde{\mathbf{X}}$ based on (9)
3. Generate diagonal matrix $\mathbf{K}_D$ with diagonal entries
   $d_{i=} \sum_{j=1}^{N} K_{i,j}$, for $i = \{1, \dots, N\}$
4. Generate $\mathbf{L}_x = \mathbf{I} - \mathbf{K}_D^{-\frac{1}{2}} \mathbf{K} \mathbf{K}_D^{-\frac{1}{2}}$
5. Generate data matrix $\hat{\mathbf{M}}$ using $\mathbf{L}$ and $\mathbf{L}_x$ based on (4)
```

$\hat{\mathbf{M}}^d$ can be readily generated based on the low rank approximation of $(\mathbf{L} + \beta \mathbf{L}_x)$ [21]. Specifically, as $(\mathbf{L} + \beta \mathbf{L}_x)$ is a PSD matrix, $(\mathbf{L} + \beta \mathbf{L}_x) = \sum_{i=1}^{N} \sigma_i \mathbf{v}_i \mathbf{v}_i^\top$ where $\mathbf{v}_i \in \mathbb{R}^N$ is the singular vector corresponding to ith singular value σ_i of $(\mathbf{L} + \beta \mathbf{L}_x)$. Based on this decomposition, the generation of data matrix $\hat{\mathbf{M}}$ follows

$$\hat{\mathbf{M}} = \operatorname{diag}\left(\frac{1}{\sqrt{\sigma_1}}, \dots, \frac{1}{\sqrt{\sigma_N}}\right) [\mathbf{v}_1, \dots, \mathbf{v}_N]^\top .$$

Similarly, $\hat{\mathbf{M}}^d$ can be created based on the best rank-d approximation of $(\mathbf{L} + \beta \mathbf{L}_x)^\dagger$ such that

$$\hat{\mathbf{M}}^d = \operatorname{diag}\left(\frac{1}{\sqrt{\sigma_1}}, \dots, \frac{1}{\sqrt{\sigma_d}}\right) [\mathbf{v}_1, \dots, \mathbf{v}_d]^\top ,$$

where $d \ll N$. If $\sqrt{\sigma_1}, \dots, \sqrt{\sigma_d}$ are the smallest d singular values of $(\mathbf{L} + \beta \mathbf{L}_x)$, the $(\hat{\mathbf{M}}^d)^\top \hat{\mathbf{M}}^d$ is the best rank-d approximation of $(\mathbf{L} + \beta \mathbf{L}_x)^\dagger$ [30].

To reduce bias, the decisions should be made independent of the sensitive attributes, which cannot be provided by only removing the sensitive attributes, as some of the nodal features and graph topology may be correlated with them [31]. Motivated by this, we aim to "filter-out" the information coming indirectly from the sensitive attributes by modifying the dimensionality reduction step. To this end, our modification design is based on the filtering operation enabled by the Graph Fourier transform (GFT) [32], which is a graph signal processing tool where input graph signals are projected onto a space spanned by the orthogonal singular/eigen-vectors of the PSD graph Laplacian matrix. Let the SVD of graph Laplacian be $\mathbf{L} = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^\top$, the GFT of input graph signal $\boldsymbol{\gamma} \in \mathbb{R}^N$ is; $\tilde{\boldsymbol{\gamma}} = \mathbf{V}^\top \boldsymbol{\gamma}$ (if the singular/eigen-vectors and graph signal $\boldsymbol{\gamma}$ are standardized before GFT, this operation represents linear correlation). In this transformation, eigenvalues of the Laplacian (a measure of smoothness over graph) resemble the frequencies in Fourier transform, therefore GFT of an input signal shows the distribution of the signal over different graph modes (eigenvectors of Laplacian).

Specifically, the correlations ρ_i between the sensitive attributes $\mathbf{s}$ and singular vectors of the graph Laplacian $\mathbf{v}_i$ are utilized to design the filter. Based on the obtained ρ_i values, the singular vectors for which the magnitudes of ρ_i are high are avoided to be used as base vectors in the dimensionality reduction step. Note that Spearman correlation (denoted by corr$(\cdot, \cdot)$, [33]) is utilized in this method instead of a linear correlation to capture the nonlinear dependencies, thus the GFT of the standardized sensitive attribute vector is not exactly used. Let $\{\sigma_1, \dots, \sigma_{d+d_s}\}$ be the set of smallest $d + d_s$ singular values of $(\mathbf{L} + \beta \mathbf{L}_x)$ with corresponding singular vectors $\{\mathbf{v}_1, \dots, \mathbf{v}_{d+d_s}\}$. Proposed fairness enhancement method obtains $\rho_i = \text{corr}(\mathbf{s}, \mathbf{v}_i)$ for $i = \{1, \dots, d + d_s\}$, and selects d singular vectors corresponding to the d smallest values among the $|\rho_i|$s. Let $\hat{\mathbf{V}} \in \mathbb{R}^{d \times N}$ denote the matrix of singular vectors resulted in the smallest absolute correlation values, and $\hat{\mathbf{\Sigma}} \in \mathbb{R}^{d \times d}$ represents a diagonal matrix with singular values corresponding to the singular vectors in $\hat{\mathbf{V}}$ as diagonal entries. Based on the created matrices $\hat{\mathbf{V}}$ and $\hat{\mathbf{\Sigma}}$, fairness-aware dimensionality reduction can be executed where $\hat{\mathbf{M}}^d_{fair} = \hat{\mathbf{\Sigma}}^{-\frac{1}{2}} \hat{\mathbf{V}} \in \mathbb{R}^{d \times N}$. Herein, the main goal is to utilize the orthogonal bases that are less correlated with the sensitive attributes in the generation of data matrix $\hat{\mathbf{M}}^d_{fair}$ without losing significant amount of information. Note that the selection of the values d and d_s provides a trade-off between the classification accuracy and fairness metrics.

Algorithm 2: Fairness-aware Dimensionality Reduction

```
Data: $\mathbf{L}, \mathbf{L}_x, \mathbf{s}, \beta, d, d_s$
Result: $\hat{\mathbf{M}}^d_{fair}$
1. Obtain singular values $\mathcal{S} = \{\sigma_1, \dots, \sigma_N\}$ and
   corresponding singular vectors $\mathcal{V} = \{\mathbf{v}_1. \dots, \mathbf{v}_N\}$ of
   matrix $(\mathbf{L} + \beta \mathbf{L}_x)$
2. Create a set from the $d + d_s$ smallest singular values
   $\mathcal{S}_{min} = \{\sigma_1, \dots, \sigma_{d+d_s}\}$ and corresponding singular
   vectors $\mathcal{V}_{min} = \{\mathbf{v}_1, \dots, \mathbf{v}_{d+d_s}\}$
3. Obtain $\rho_i = \text{corr}(\mathbf{s}, \mathbf{v}_i)$ for
   $\forall \mathbf{v}_i \in \mathcal{V}_{min}, i = \{1, \dots, d + d_s\}$
4. Generate $\hat{\mathbf{V}} \in \mathbb{R}^{d \times N}$ by selecting $d$ of the $\mathbf{v}_i$s that result
   in $d$ smallest $|\rho_i|$ values
5. Generate diagonal matrix $\hat{\mathbf{\Sigma}} \in \mathbb{R}^{d \times d}$ with singular values
   corresponding to singular vectors in $\hat{\mathbf{V}}$ as diagonal entries
5. Generate $\hat{\mathbf{M}}^d_{fair} \in \mathbb{R}^{d \times N}$ such that $\hat{\mathbf{M}}^d_{fair} = \hat{\mathbf{\Sigma}}^{-\frac{1}{2}} \hat{\mathbf{V}}$
```

Note that regardless of employing our fairness enhancement method, dimensionality reduction is built upon singular value decomposition (SVD). Assuming standard matrix multiplication, the computational complexity of SVD is $\mathcal{O}(N^3)$. Our proposed fairness enhancement scheme involves $d + d_s$ correlation calculations in between vectors in $\mathbb{R}^N$, which introduces an additional $\mathcal{O}\left(N(d + d_s)\right)$. In addition, picking the d smallest elements among $d + d_s$ elements incurs $\mathcal{O}\left(d(d + d_s)\right)$. Since $d \ll N$, these overall additive complexities are negligible when compared to $\mathcal{O}(N^3)$ of SVD, suggesting that the proposed fairness enhancement method incurs low extra complexity (negligible as $N \to \infty$) over the standard dimensionality reduction procedure it is built upon.

3. EXPERIMENTAL RESULTS

3.1. Data Set and Experimental Setup

In the experiments, two datasets Pokec-z and Pokec-n [27] generated from a real social network, Pokec [34], are used. Pokec is a Facebook-like social network used in Slovakia and the utilized data is sampled from the whole network of 2012 in an anonymized way [34]. Pokec-z and Pokec-n are created by collecting the information of users from two major regions [27]. The sensitive attribute of the users is selected to be the region information, while the label is the working field. Both sensitive attributes and labels are binarized as in [27]. In the experiments, graph nodes are the users in the social network, and the information of $N = 7,659$ and $6,185$ users are utilized in Pokec-z and Pokec-n datasets, respectively. The number of nodal features is $F = 59$ for both datasets.

While the presented method in this study for the use of nodal features can be applied to any online learning and selective sampling algorithms on graphs based on the learning problem in (3), three of

Table 1. Results for the proposed nodal feature fusion algorithms.

	Pokec-z				Pokec-n			
	ER (%)	Δ_{SP} (%)	Δ_{EO} (%)	Q_{num}	ER (%)	Δ_{SP} (%)	Δ_{EO} (%)	Q_{num}
OLLGC	41.69	4.33	6.84	-	42.11	6.49	6.57	-
SSLGC	39.91	10.00	4.65	2622	40.61	10.69	8.86	2173
AGS	38.67	11.67	14.53	3865	39.37	11.30	9.56	3077
OLLGC, Direct Fusion	39.77	5.59	6.98	-	41.63	4.99	4.39	-
SSLGC, Direct Fusion	38.65	3.74	4.87	5054	41.44	5.22	2.94	4189
AGS, Direct Fusion	35.64	8.72	9.67	2680	38.87	6.95	5.46	2309
OLLGC, Kernel Fusion	39.23	7.54	10.81	-	40.99	7.03	8.76	-
SSLGC, Kernel Fusion	37.12	5.75	8.20	2563	39.25	2.34	4.35	1849
AGS, Kernel Fusion	35.58	14.49	16.86	3617	37.05	10.70	12.67	2912

Table 2. Results after proposed fairness enhancement.

	Pokec-z				Pokec-n			
	ER (%)	Δ_{SP} (%)	Δ_{EO} (%)	Q_{num}	ER (%)	Δ_{SP} (%)	Δ_{EO} (%)	Q_{num}
Fair-OLLGC	42.60	7.26	1.58	-	43.30	5.33	5.37	-
Fair-SSLGC	40.93	7.64	0.58	2636	42.11	9.46	4.12	2437
Fair-AGS	39.48	10.04	3.65	3827	40.25	5.91	3.74	3222
Fair-OLLGC, Kernel Fusion	40.23	2.42	4.12	-	40.94	1.95	2.97	-
Fair-SSLGC, Kernel Fusion	37.64	1.84	5.40	2064	39.47	1.19	2.42	2631
Fair-AGS, Kernel Fusion	36.34	5.48	2.14	3500	37.67	7.77	6.52	3009

the pioneering studies are selected as baselines. Baselines include online learning with local and global consistency (OLLGC) [21], selective sampling with local and global consistency (SSLGC) [21], and aggressive graph-based selective sampling (AGS) [22]. For the baseline algorithms, OLLGC, SSLGC, and AGS, the parameter μ is tuned by searching the grid $\{10^{-3}, 10^{-2}, \ldots, 10\}$ on a held-out set, while all other parameters are fixed to the values that are selected in the corresponding studies (for OLLGC, SLLGC $\kappa = 0.4$, and for AGS $\gamma = 1, h = 0.01$). Additionally, β in (4) and d_s are tuned by searching the grids $\{10^{-1}, 1, 10\}$, and $\{5, 10, 25\}$ on a held-out set, respectively. In all experiments, d is chosen to be 100.

3.2. Results

In the experiments, cumulative error rate (ER) of the online algorithms is used as the performance metric. Furthermore, the fairness evaluation for the algorithms is made in terms of **statistical parity**: $\Delta_{SP} = |P(\hat{y} = 1 \mid s = 0) - P(\hat{y} = 1 \mid s = 1)|$ and **equal opportunity**: $\Delta_{EO} = |P(\hat{y} = 1 \mid y = 1, s = 0) - P(\hat{y} = 1 \mid y = 1, s = 1)|$, where y denotes the ground truth label, and $\hat{y}$ is the predicted label. Smaller values for Δ_{SP} and Δ_{EO} are desired. The overall results are generated by shuffling the data 3 times and taking average of the results. In the presented tables, Q_{num} denotes the total number of queried labels.

Fusion of the Nodal Features: The results for the algorithms that utilize nodal features are presented in Table 1 together with the results for baseline algorithms with no nodal information. As a baseline for the utilization of nodal features, the result of the concatenated nodal features $\mathbf{x}$ with data vectors $\mathbf{m}$ is also presented. Note that, to keep the dimension of the resulted fusion vector the same as other methods, 50 dimensional feature vectors obtained via principal component analysis are concatenated with 50 dimensional $\mathbf{m}$ vectors for which d is selected as 50 in dimensionality reduction. This method is called *Direct Fusion*, while the presented approach in this study is referred to as *Kernel Fusion*.

The results demonstrate that utilizing nodal features together with the proposed method can consistently improve performances of the baselines by approximately $2 - 3\%$. Furthermore, this error rate improvement can be achieved with less number of queried labels as well. Additionally, the superior performance of Kernel Fusion over Direct Fusion in terms of cumulative error rates demonstrates the gain of the utilized fusion scheme over naive concatenation. The results of Table 1 show another noteworthy phenomenon: While the employment of nodal features through Kernel Fusion can help with error rates, utilizing nodal information propagates the bias within them and causes worse results in fairness.

Fairness: The results for the proposed bias-reduction method are presented in Table 2. We note that the prefix "Fair" denotes the fairness aware version of the scheme whose name it precedes. The results of Table 2 show the overall effectiveness of the proposed fairness enhancement method for both original algorithms and their improved versions that utilize nodal features. It is observed that fairness improvements are larger when the nodal attributes are also used. Specifically both fairness measures are reduced by approximately 50% for each algorithm that utilizes nodal features. Overall, the presented results illustrate the power of the proposed method in the reduction of statistical parity and equal opportunity without incurring a significant additional complexity.

4. CONCLUSION

In this study, it has been shown that the performance of the existing online learning and selective sampling algorithms on graphs can be improved with the utilization of nodal features. Experimental results on real social networks show that the inclusion of nodal attributes leads to an approximately $2 - 3\%$ error rate decrease for both online learning and selective sampling algorithms, though with amplified bias due to the propagated bias by the nodal features. To address the fairness issue, a method that can be used together with the dimensionality reduction step of the original algorithms is proposed, and its efficacy has been demonstrated in terms of the statistical parity and equal opportunity. The proposed bias reduction method can be employed in any graph signal processing application, and a future work of this study is to investigate its effect in a more general setting.

5. REFERENCES

[1] A. Sala, L. Cao, C. Wilson, R. Zablit, H. Zheng, and B. Y. Zhao, "Measurement-calibrated graph models for social network experiments," in *Proc. Int. Conf. on World Wide Web (WWW)*, Apr. 2010, pp. 861–870.

[2] G. A. Pagani and M. Aiello, "The power grid as a complex network: a survey," *Physica A: Statistical Mechanics and its Applications*, vol. 392, no. 11, pp. 2688–2700, 2013.

[3] B. H. Voy, J. A. Scharff, A. D. Perkins, A. M. Saxton, B. Borate, E. J. Chesler, L. K. Branstetter, and M. A. Langston, "Extracting gene networks for low-dose radiation using graph theoretical algorithms," *PLoS Computational Biology*, vol. 2, no. 7, p. e89, 2006.

[4] O. Chapelle, B. Schlkopf, and A. Zien, *Semi-Supervised Learning*, 1st ed. The MIT Press, 2010.

[5] A. Shen, R. Tong, and Y. Deng, "Application of classification models on credit card fraud detection," in *Int. Conf. on Service Systems and Service Management*. IEEE, June 2007, pp. 1–4.

[6] S. Parisot, S. I. Ktena, E. Ferrante, M. Lee, R. Guerrero, B. Glocker, and D. Rueckert, "Disease prediction using graph convolutional networks: application to autism spectrum disorder and alzheimer's disease," *Medical Image Analysis*, vol. 48, pp. 117–130, 2018.

[7] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proc. Int. Conf. on Machine learning (ICML)*, Aug. 2003, pp. 912–919.

[8] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," *Adv. in Neural Information Processing Systems (NIPS)*, vol. 16, no. 16, pp. 321–328, Dec. 2004.

[9] M. Herbster and M. Pontil, "Prediction on a graph with a perceptron," in *Adv. in Neural Information Processing Systems (NIPS)*, vol. 21, Dec. 2006, pp. 577–584.

[10] M. Herbster and G. Lever, "Predicting the labelling of a graph via minimum p-seminorm interpolation," in *Adv. in Neural Information Processing Systems (NIPS)*, Dec. 2009.

[11] N. Slonim *et al.*, "Active online classification via information maximization," in *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, vol. 22, no. 1, July 2011.

[12] P. Zhao and S. C. Hoi, "Cost-sensitive online active learning with application to malicious url detection," in *Proc. ACM Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, Aug. 2013, pp. 919–927.

[13] J. Lu, P. Zhao, and S. Hoi, "Online passive aggressive active learning and its applications," in *Asian Conf. on Machine Learning*, Nov. 2015, pp. 266–282.

[14] M. Ji and J. Han, "A variance minimization criterion to active learning on graphs," in *Artificial Intelligence and Statistics (AISTAT)*, Apr. 2012, pp. 556–564.

[15] Q. Gu and J. Han, "Towards active learning on graphs: An error bound minimization approach," in *IEEE Int. Conf. on Data Mining*, Dec. 2012, pp. 882–887.

[16] N. Cesa-Bianchi, C. Gentile, F. Vitale, and G. Zappella, "Active learning on trees and graphs," in *Annual Conference on Learning Theory*, June 2010, pp. 320–332.

[17] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby, "Selective sampling using the query by committee algorithm," *Machine learning*, vol. 28, no. 2, pp. 133–168, 1997.

[18] F. Orabona and N. Cesa-Bianchi, "Better algorithms for selective sampling," in *Proc. Int. Conf. on Machine Learning (ICML)*, July 2011, pp. 433–440.

[19] N. Cesa-Bianchi, C. Gentile, and F. Orabona, "Robust bounds for classification via selective sampling," in *Proc. Int. Conf. on Machine Learning (ICML)*, June 2009, pp. 121–128.

[20] M. Herbster and M. Pontil, "Prediction on a graph with a perceptron," in *Adv. in Neural Information Processing Systems (NIPS)*, vol. 21. Citeseer, Dec. 2006, pp. 577–584.

[21] Q. Gu, C. Aggarwal, J. Liu, and J. Han, "Selective sampling on graphs for classification," in *Proc. ACM Int. Conf. on Knowledge Discovery and Data Mining (KDD)*, Aug. 2013, pp. 131–139.

[22] P. Yang, P. Zhao, V. W. Zheng, and X.-L. Li, "An aggressive graph-based selective sampling algorithm for classification," in *IEEE Int. Conf. on Data Mining*, Nov. 2015, pp. 509–518.

[23] A. Beutel, J. Chen, Z. Zhao, and E. H. Chi, "Data decisions and theoretical implications when adversarially learning fair representations," *arXiv preprint arXiv:1707.00075*, 2017.

[24] J. Buolamwini and T. Gebru, "Gender shades: Intersectional accuracy disparities in commercial gender classification," in *Conf. on Fairness, Accountability and Transparency*, Feb. 2018, pp. 77–91.

[25] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, "Fairness through awareness," in *Proc. Innovations in Theoretical Computer Science Conf.*, Jan. 2012, pp. 214–226.

[26] T. A. Rahman, B. Surma, M. Backes, and Y. Zhang, "Fairwalk: Towards fair graph embedding." in *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, July 2019, pp. 3289–3295.

[27] E. Dai and S. Wang, "Say no to the discrimination: Learning fair graph neural networks with limited sensitive attribute information," in *Proc. the ACM International Conference on Web Search and Data Mining*, Mar. 2021, pp. 680–688.

[28] R. K. Ando and T. Zhang, "Learning on graph with laplacian regularization," *Adv. in Neural Information Processing Systems (NIPS)*, vol. 19, p. 25, Dec. 2007.

[29] S. Shalev-Shwartz *et al.*, "Online learning and online convex optimization," *Foundations and Trends in Machine Learning*, vol. 4, no. 2, pp. 107–194, 2011.

[30] G. H. Golub and C. F. Van Loan, *Matrix Computations (3rd Ed.)*, 1996.

[31] S. Hajian and J. Domingo-Ferrer, "A methodology for direct and indirect discrimination prevention in data mining," *IEEE Trans. on Knowledge and Data Engineering*, vol. 25, no. 7, pp. 1445–1459, July 2013.

[32] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[33] D. Zwillinger and S. Kokoska, *CRC standard probability and statistics tables and formulae*. Crc Press, 1999.

[34] L. Takac and M. Zabovsky, "Data analysis in public social networks," in *Int. Scientific Conf. and Int. Workshop. 'Present Day Trends of Innovations'*, vol. 1, no. 6, May 2012.