SPARC: Self-Paced Network Representation for Few-Shot Rare Category Characterization

Dawei Zhou[†], Jingrui He[†], Hongxia Yang[‡], and Wei Fan[††]
[†]Arizona State University, {dzhou23, jingrui.he}@asu.edu;
[‡]Alibaba Group, hongxia.yang1@gmail.com;
[††]Tencent Medical AI Lab, davidwfan@tencen.com

ABSTRACT

In the era of big data, it is often the rare categories that are of great interest in many high-impact applications, ranging from financial fraud detection in online transaction networks to emerging trend detection in social networks, from network intrusion detection in computer networks to fault detection in manufacturing. As a result, rare category characterization becomes a fundamental learning task, which aims to accurately characterize the rare categories given limited label information. The unique challenge of rare category characterization, i.e., the non-separability nature of the rare categories from the majority classes, together with the availability of the multi-modal representation of the examples, poses a new research question: how can we learn a salient rare category oriented embedding representation such that the rare examples are well separated from the majority class examples in the embedding space, which facilitates the follow-up rare category characterization?

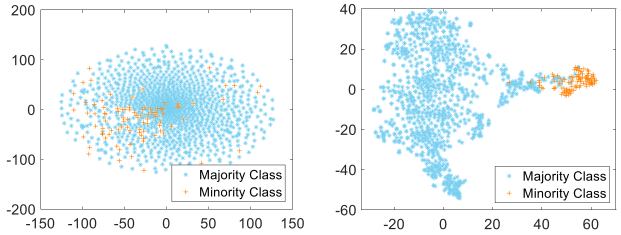
To address this question, inspired by the family of curriculum learning that simulates the cognitive mechanism of human beings, we propose a self-paced framework named *SPARC* that gradually learns the rare category oriented network representation and the characterization model in a *mutually beneficial* way by shifting from the 'easy' concept to the target 'difficult' one, in order to facilitate more reliable label propagation to the large number of unlabeled examples. The experimental results on various real data demonstrate that our proposed *SPARC* algorithm: (1) shows a significant improvement over state-of-the-art graph embedding methods on representing the rare categories that are non-separable from the majority classes; (2) outperforms the existing methods on rare category characterization tasks.

KEYWORDS

Rare Category Analysis, Network Embedding, Self-Paced Learning

ACM Reference Format:
Dawei Zhou[†], Jingrui He[†], Hongxia Yang[‡], and Wei Fan[††]. 2018. SPARC: Self-Paced Network Representation for Few-Shot Rare Category Characterization. In *ACM SIGKDD, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3219819.3219952

KDD 2018, August 19–23, 2018, London, United Kingdom
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219952



(a) Original Feature Space (b) Embedding Space

Figure 1: Rare category oriented network representation: the majority and minority classes are not separable in the original feature space, but become well separated in the embedding space induced by *SPARC*.

1 INTRODUCTION

In many real-world applications, it is usually the case that the rare categories play an essential role despite their extreme scarcity. For example, in transaction networks, the vast majority of online transactions are legitimate, and only a small number may be fraudulent; in social networks, the majority users could be loss of sight to the underlying emerging trends, which could potentially turn into a burst in the near future; in computer networks, the percentage of network intrusion among the huge volumes of routine network traffic is small, but the loss might be significant.

One key challenge for analyzing the rare categories is the non-separable nature, i.e., the support regions of majority and minority in networks are usually non-separable. For example, in the financial fraud detection, the fraudulent people often try to camouflage their synthetic identities within the normal ones in order to bypass the fraud detection systems [9]; in the spam detection, the junk mails are deliberately made like the normal ones [18]. In addition, due to the highly skewness and non-separable nature of rare categories, labeling rare category examples is extremely expensive. In the extreme case, we may need to train the rare category analysis model from very few or only one labeled example. That said, it is therefore a very important and challenging task to identify such minority classes given that they are (1) highly skewed, (2) non-separable and (3) sparsely labeled. To be more specific, in this paper, we want to answer the following two open questions: First (*T1. Embedding*), how to learn a salient rare category oriented embedding representation in order to better characterize them when the minority classes are non-separable from the majority classes? Second (*T2. Characterization*) how to accurately characterize the rare examples in the scarcity of label information?

Recently developed network embedding techniques [10, 31, 37], that encode graph structural information into a low dimensional representation, have received much success in boosting the performance of various network interface capabilities such as entity classification [44], author identification [7] and community detection [39]. However, these network embedding models are usually trained by uniformly drawing graph context without considering the scenario that the networks may exhibit imbalanced class distribution. Thus, the context information of rare categories may not be well preserved in the extracted training context pairs by existing context sampling methods [10, 31, 37, 44], which could be a key issue in the follow-up rare category characterization.

To counter the negative effects from learning in an imbalanced data set, extensive deep models [32, 34] have been proposed based on the re-sampling strategy [6], the cost sensitive learning [45] or adapting learning [12]. However, in the rare category characterization setting, training the aforementioned deep models in the scarcity of labeled rare category examples often suffers from the inevitable errors during label propagation. Thus, how to maintain a 'safe and secure' label propagation is of the key importance in learning the underlying distribution of rare categories.

To address the above challenges, in this paper, we propose a generic rare category analysis framework named *SPARC*, that jointly predict the rare category examples and the neighborhood context in the graph. Our proposed *SPARC* is designed to jointly address two tasks, namely *T1. Embedding* and *T2. Characterization*, in a mutually beneficial way. In order to alleviate the influence of ambiguous data during model training, we integrate the self-paced learning paradigm into our framework to jointly select the rare category oriented graph contexts and maintain a reliable label propagation for training our proposed *SPARC* model.

The main contributions of this paper are summarized below.

(1) **Problem.** We formalize the problems of rare category oriented network representation and characterization learning in attributed networks, and identify their unique challenges from the nature of rare categories.
(2) **Algorithms.** We propose a generic rare category analysis framework named *SPARC*, which is able to jointly predict the rare category examples and the neighborhood context in the attributed network.
(3) **Evaluations.** Extensive experimental results on real networks demonstrate the performance of the proposed *SPARC* algorithm.

The rest of our paper is organized as follows. Related works are reviewed in Section 2, followed by the notation and problem definition in Section 3. In Section 4, we present our proposed framework *SPARC*. Experimental results are reported in Section 5 before we conclude the paper in Section 6.

2 RELATED WORK

In this section, we briefly review the related works regarding rare category analysis, network representation and curriculum learning.

2.1 Rare Category Analysis

Different from outlier detection [15, 25, 26] that targets to find abnormal patterns that do not conform to the expectation, and imbalanced classification [6] that aims to increase the overall accuracy, rare category analysis explores the compactness of the minorities and characterizes them from the highly skewed data sets. Rare category analysis (RCA) is first introduced by Pelleg and Moore [30], where the rare categories are defined as the minority clusters that exhibit a compact property in an imbalanced data distribution. The unique challenges of RCA come from the highly skewed data distribution, together with the non-separability nature of the rare categories from the majority classes. Up until now, researchers have proposed various methods for the RCA problem, such as sampling-based methods [6, 13, 47], ensemble-based methods [36], algorithm-adaptation-based methods [40], and maximum-margin-based methods [14]. Recently, [16] presented a deep representation model for the imbalanced data by enforcing the deep model to explore and maintain the inter-cluster and inter-class margins. [51] proposed a local graph clustering algorithm that identifies the structure-rich clusters by exploring the high-order structures in the neighborhood of the initial vertex in the given graph. However, very little work (if any) is devoted to learning a rare category oriented graph representation in the class-imbalanced networks. In this paper, we propose a rare category oriented network embedding approach, which jointly leverages the neighbored context information and the label information of rare examples, in order to better characterize the rare categories in the embedding space.

2.2 Network Representation

The pioneer works of graph representation can be traced back to the early 2000s, when many methods [1, 20, 33, 38] were developed for learning a low-dimensional graph representation with a minimized reconstruction error. While the network interface abilities of these methods may suffer from overfitting or poor scalability in real applications [7, 39]. Recently, a surge of research interests on network embedding by employing Skipgram model [29] has been observed in the network science. Among them, DeepWalk [31] firstly generalizes the Skipgram model to embed the graph context in a low-dimensional representation, where the graph context is extracted based on a truncated random walk; LINE [37] further extends the model by introducing an optimized objective function that incorporates the first-order and the second-order proximities to learn network representation; node2vec [10] preserves both homophily and structural equivalence relationships by generating the graph context with a biased random walk. In spite of the general-purposed network embedding approaches, a diversity of researches have been conducted to learn network representations for solving specific tasks with training examples or prior knowledge, such as multi-network inferences [23], author identification [7], entity classification [24, 44] and community detection [39]. Despite the success of these methods, embedding representation of class-imbalanced networks has heretofore received little attention. In this paper, we aim to learn a salient rare category oriented embedding representation, such that the minority classes are well separated from the majority classes, which facilitates the follow-up rare category analysis tasks such as detection [8, 13, 49, 50], prediction [11], clustering [46, 51] and classification [14, 41, 48].

2.3 Curriculum Learning

Inspired by the cognitive process of humans, Bengio's group proposes the curriculum learning (CL) paradigm, in which the underlying model is gradually trained from easy aspects of a task to the complex ones based on the predetermined 'curriculum' [2, 3]. This theory has been successfully applied to various applications, such as geometrical shape classification [3], teaching a robot of the concept of 'graspability' [19], grammar induction [35], etc. However, the heuristical curriculum design in CL turns out onerous or conceptually difficult in many real problems [21]. To eliminate this issue, Kumar et al. [21] propose a new learning paradigm named self-paced learning (SPL), which automatically learns a 'curriculum' by minimizing the loss function with a self-paced regularizer. In particular, SPL jointly updates the model parameters $\boldsymbol{w}$ and the 'curriculum' indicator variable $\boldsymbol{v}$ by optimizing the following objective:

$$\min_{\boldsymbol{w},\boldsymbol{v}} \sum_i v_i \mathbb{L}(y_i, f(\boldsymbol{x_i}, \boldsymbol{w})) - \lambda \sum_i v_i, \quad s.t. \boldsymbol{v} \in [0,1]^n, \tag{1}$$

where $\mathbb{L}(y_i, f(\boldsymbol{x_i}, \boldsymbol{w}))$ denotes the loss function, and λ is the self-paced parameter for controlling the learning pace. BCU [43] (Block-Coordinate Update) is usually adopted to solve the above bi-convex optimization problem by dividing the variables into disjoint blocks and alternatively optimizing one block while keeping the rest fixed. More recently, in [17], the authors develop a unified framework that improves CL and SPL by considering both the prior knowledge and the learning progress during training; in [27], the authors propose a self-paced co-training algorithm, which is proved to guarantee the theoretical effectiveness under the ϵ-expansion assumption. In this paper, we advance the SPL scheme to the scenario of rare category analysis in the scarcity of labeled example, in order to gradually learn the rare category oriented network representation and the characterization model in a mutually beneficial way.

3 PROBLEM DEFINITION

Throughout the paper, we use lowercase letters to denote scalars (e.g., α), boldface lowercase letters to denote vectors (e.g., $\boldsymbol{v}$), and boldface uppercase letters to denote matrices (e.g., $\boldsymbol{A}$). Following the convention in Matlab, we represent the i^{th} row of matrix $\boldsymbol{A}$ as $\boldsymbol{A(i,:)}$, the j^{th} column of matrix $\boldsymbol{A}$ as $\boldsymbol{A(:,j)}$, the entry of the i^{th} row and the j^{th} column in matrix $\boldsymbol{A}$ as $\boldsymbol{A(i,j)}$, and the transpose of matrix $\boldsymbol{A}$ as $\boldsymbol{A}^T$. Given an attributed network $G = (\boldsymbol{V}, \boldsymbol{E}, \boldsymbol{X})$, where $\boldsymbol{V}$ consists of n vertices, $\boldsymbol{E}$ consists of m edges, and $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x_n}\} \in \mathbb{R}^{n\times r}$ denotes the set of nodes' attributes, we use $\boldsymbol{A}$ to represent the adjacency matrix of G. Let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_L \in \mathbb{R}^r$ denote the L labeled examples, where we assume there is at least one from each minority class; let $\boldsymbol{x}_{L+1}, \ldots, \boldsymbol{x}_{L+U} \in \mathbb{R}^r$, where $n = L + U$, denote the U unlabeled examples, which either come from the majority class, i.e., $y_i \in \{0\}$, or the $c \geq 1$ minority classes, i.e., $y_i \in \{1, \ldots, c\}$. With the above notation, our problem can be formally defined as follows:

Problem 1. *Rare Category Embedding Representation (RCE)*

Input: *(i) an attributed network $G = (V, E, \boldsymbol{X})$, (ii) one-shot or few-shot labeled examples $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_L$, and (iii) the desired embedding dimension d.*

Output: *a d-dimensional embedding representation $\boldsymbol{E} \in \mathbb{R}^{n\times d}$ that preserves the underlying structure and context information, especially for the rare categories.*

The output of Problem 1 is a low-dimensional matrix $\boldsymbol{E}$, where the i^{th} row (i.e., a d-dimensional vector $\boldsymbol{e}_i$) encodes the discriminative attributes and topology context information of node i that are beneficial for characterizing rare categories. The premise of network embedding models is to preserve different types of proximities between vertices and their neighborhood in a semi-supervised, e.g., [44], or unsupervised manner, e.g., [10, 31, 37]. However, the existing methods are not best suited for characterizing the rare categories, which are (1) under-represented in the given network, (2) non-separable from the majority classes, and (3) provided with scarce labeled examples in a massive attributed network. Here, we aim to learn a rare category oriented embedding representation that can incorporate the label and context information to better characterize the minority classes.

Problem 2. *Rare Category Characterization (RCC)*

Input: *(i) an attributed network $G = (V, E, \boldsymbol{X})$, and (ii) one-shot or few-shot labeled examples $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_L$.*

Output: *a list of predicted rare category examples.*

The main challenges of Problem 2 come from the highly skewed class membership and the scarce training data. Due to these issues, the existing imbalanced classification algorithms and semi-supervised learning techniques may suffer from overfitting and inevitable errors in label propagation. Notice that Problem 1 and Problem 2 are related with one another, and may be mutually beneficial if jointly solved in the sense that (1) incorporating the rare category oriented graph context information that is preserved in RCE is crucial for characterizing the rare examples in Problem 2, and (2) the trained RCC model could serve as a 'supervisor' to determine the rare category oriented graph context for learning the network representation in Problem 1. Due to these reasons, we present a generic rare category analysis framework in the following section, which is capable to learn from a handful or even one-shot training example and maintain a 'safe and secure' label propagation process in order to jointly address Problem 1 and Problem 2.

4 PROPOSED MODEL

In this section, we present our rare category analysis framework *SPARC*, which simultaneously learns the graph embedding and predicts the rare category examples in a mutually beneficial way. We first formulate it as a generic optimization problem, and then present the details on how to jointly learn a rare category oriented embedding and characterize rare category examples within a self-paced learning paradigm.

4.1 A Generic Joint Learning Framework

To address the proposed RCE and RCC problems, our joint learning framework should take into consideration the following key aspects. First (*skewed distribution*), in order to detect and characterize the rare categories, our joint learning framework should have the capability to model the imbalanced class memberships in the given networks. Second (*non-separability*), the minority classes and majority classes are often non-separable in both the network

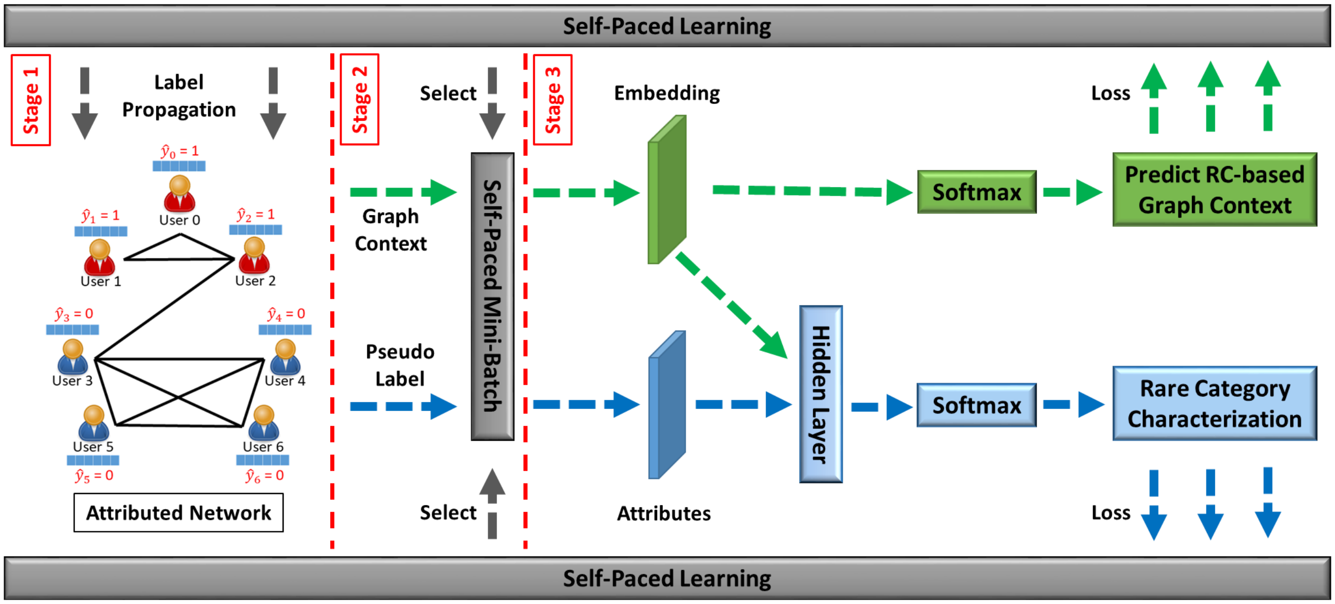


Figure 2: Illustration of the proposed *SPARC* framework. The minority class examples and the majority class examples are represented by the red and blue icons, respectively. In the given networks, only one minority class example (i.e., User 0) is labeled, while the remaining nodes are iteratively assigned with pseudo labels $\hat{y}$ by the rare category characterization model, in order to learn the underlying distribution of the rare category.

topology space (i.e., $\boldsymbol{A}$) and the feature space (i.e., $\boldsymbol{X}$) . Therefore, rare category oriented representation should result in the minority examples being largely separated from the majority classes in the embedding space. Third (*label scarcity*), due to the hardness and expensive cost of labeling rare category examples, our proposed framework should be capable to learn from few shot or even only one labeled rare category example.

We start by illustrating our framework in the binary case with only one majority class and one minority class in the given network. The extension to multi-class RCC problem will be discussed later in Subsection 4.2. With these design objectives in mind, we propose a generic rare category analysis framework as an optimization problem with the following objective function:

$$\begin{aligned}\mathcal{L}_b =& \mathcal{L}_s + \mathcal{L}_{rc} + \mathcal{L}_{tc} + \mathcal{L}_{sp} + \mathcal{L}_{co}\\ &= \underbrace{\sum_{i=1}^{L} c_{y_i,\hat{y}_i} \log Pr(\hat{y}_i = 1 - y_i | \boldsymbol{x_i}, \boldsymbol{e_i})}_{\mathcal{L}_s\text{: cost sensitive learning}}\\ &\quad \underbrace{- \sum_{i=1}^{L+U} v_i^{(1)} \log Pr(\hat{y}_i = 1 | \boldsymbol{x_i}, \boldsymbol{e_i})}_{\mathcal{L}_{rc}\text{: predict rare category examples}} \underbrace{- \sum_{i=1}^{L+U} v_i^{(0)} E_{(i,c,\gamma)} \log \sigma(\gamma \boldsymbol{\theta_c^T} \boldsymbol{e_i})}_{\mathcal{L}_{tc}\text{: predict graph context}}\\ &\quad \underbrace{- \sum_{i=1}^{L+U} \lambda^{(1)} v_i^{(1)} + \lambda^{(0)} v_i^{(0)}}_{\mathcal{L}_{sp}\text{: self-paced regularizer}} \quad \underbrace{-\alpha \sum_{i=1}^{L+U} v_i^{(1)} v_i^{(0)}}_{\mathcal{L}_{co}\text{: consensus regularizer}} \qquad (2)\end{aligned}$$

where the objective function consists of five terms. The first term $\mathcal{L}_s$ is the cost sensitive loss over the labeled data, in which $c_{y_i,\hat{y}_i}$ denotes the misclassification cost of labeling node i belonging to class y_i into a different class $\hat{y}_i \neq y_i$. In particular, we let $c_{1,0} > c_{0,1} \geq 1$ in order to further penalize the errors of classifying the minority class examples into the majority class. The second term $\mathcal{L}_{rc}$ corresponds to the characterization step, which learns the underlying distribution of the target rare category from both labeled and unlabeled data. The third term $\mathcal{L}_{tc}$ corresponds to the embedding step, which minimizes the prediction loss regarding the sampled graph context pairs. The fourth term is the self-paced regularizer $\mathcal{L}_{sp}$, which globally maintains the learning pace of the embedding step ($\mathcal{L}_{tc}$) and the characterization step ($\mathcal{L}_{rc}$) by utilizing self-paced vectors, i.e., $\boldsymbol{v}^{(0)}, \boldsymbol{v}^{(1)} \in [0,1]^n$, respectively. The last term is the consensus regularizer $\mathcal{L}_{co}$, where α is a positive constant to balance the impact of this term on the overall objective function.

Based on Eq. 2, we propose the overall *SPARC* framework as shown in Fig. 2, where the RCE and RCC models are gradually trained in a mutually beneficial way via multiple self-paced cycles to maintain a 'safe and secure' label propagation. In particular, within each training cycle, our proposed framework *SPARC* can be decomposed into three stages. In the first stage, *SPARC* assigns the pseudo labels to the potential rare category examples based on the current prediction model. The second stage is the key step of our proposed *SPARC* model, which jointly selects the rare category oriented graph contexts and reliable predictions for training RCE and RCC models. The third stage involves the construction of two deep neural networks (DNN), including the RCE DNN (upper level) and the RCC DNN (lower level). By using the sampled graph context in Stage 2, the RCE DNN is trained to learn a salient embedding space for the RCC problem. Given the input feature vector $\boldsymbol{x_i}$ and the learned embedding vector $\boldsymbol{e_i}$, the RCC DNN is updated by learning from both the labeled and unlabeled data. In particular, the posterior probability $Pr(y_i|\boldsymbol{x_i}, \boldsymbol{e_i})$ in Eq. 2 is written as:

$$Pr(y_i|\boldsymbol{x_i}, \boldsymbol{e_i}) = \frac{exp[\boldsymbol{h^k}(\boldsymbol{x_i})^T, \boldsymbol{h^l}(\boldsymbol{e_i})^T]\boldsymbol{\theta_y}}{\sum_{y'} exp[\boldsymbol{h^k}(\boldsymbol{x_i})^T, \boldsymbol{h^l}(\boldsymbol{e})^T]\boldsymbol{\theta_{y'}}}$$

where $\boldsymbol{h^k}$ denotes the k^{th} hidden layer, and $[\cdot, \cdot]$ denotes the concatenation operator of two vectors. In the next cycle, the learned RCC DNN will be used for label propagation in Stage 1, and the learned RCE will be fed into the RCC DNN in Stage 3. To further show how *SPARC* works, we focus on the following three aspects.

Impact of the Self-Paced Learning: In the case of non-separable rare categories with scarce training data, deep discriminative models often suffer from the errors during label propagation. To address this issue, our framework exploits the SPL scheme to gradually learn from the labeled and unlabeled data, which has demonstrated its robustness in the semi-supervised setting [19, 42]. For jointly modeling the RCE and RCC problems, we design our *SPARC* framework via dual-level SPL, by leveraging the idea of co-training [4, 27]. In particular, the overall objective of *SPARC* in Eq. 2 can be interpreted as the sum of a self-paced RCE model $\mathcal{L}_{RCE}$, a self-paced RCC model $\mathcal{L}_{RCC}$ and a consensus regularizer $\mathcal{L}_{co}$ as follows:

$$\mathcal{L}_b = \mathcal{L}_{RCC} + \mathcal{L}_{RCE} + \mathcal{L}_{co}$$

where

$$\mathcal{L}_{RCC} = \mathcal{L}_s - \sum_{i=1}^{L+U} v_i^{(1)} \log Pr(\hat{y_i} = 1|\boldsymbol{x_i}, \boldsymbol{e_i}) - \sum_{i=1}^{L+U} \lambda^{(1)} v_i^{(1)} \quad (3)$$

$$\mathcal{L}_{RCE} = - \sum_{i=1}^{L+U} v_i^{(0)} E_{(i,c,\gamma)} \log \sigma(\gamma \boldsymbol{\theta_c}^T \boldsymbol{e_i}) - \sum_{i=1}^{L+U} \lambda^{(0)} v_i^{(0)} \quad (4)$$

In other words, $\mathcal{L}_{RCC}$ is mainly used in RCC DNN to address Problem 2, whereas $\mathcal{L}_{RCE}$ is mainly used in RCE DNN to address Problem 1. In addition, the consensus regularizer $\mathcal{L}_{co}$ is imposed on both $\mathcal{L}_{RCC}$ and $\mathcal{L}_{RCE}$ to ensure the 'learning curriculum' generated by *SPARC* emphasizes on learning the underlying distribution of rare categories.

We adopt BCU [43] to update the dual-level SPL in an alternative way. When we update the self-paced vector $\boldsymbol{v^{(1)}}$, the partial derivative of Eq. 2 with respect to $v_i^{(1)}$ (the i^{the} element of $\boldsymbol{v^{(1)}}$), $i = 1, \ldots, n$, can be derived as:

$$\frac{\partial \mathcal{L}_b}{\partial v_i^{(1)}} = - \log Pr(\hat{y_i} = 1|\boldsymbol{x_i}, \boldsymbol{e_i}) - \lambda^{(1)} - \alpha v_i^{(0)} \quad (5)$$

Thus, the closed-form solution to update $v_i^{(1)}$ is

$$v_i^{(1)} = \begin{cases} 1 & - \log Pr(\hat{y_i} = 1|\boldsymbol{x_i}, \boldsymbol{e_i}) < \lambda^{(1)} + \alpha v_i^{(0)} \\ 0 & \text{Otherwise} \end{cases} \quad (6)$$

By updating self-paced vector $\boldsymbol{v^{(1)}}$, we can identify the reliable predictions in order to learn the underlying distribution of rare category in RCC DNN. To be specific, given the self-paced parameter $\lambda^{(1)}$, examples with a higher confidence to belong to the minority class, i.e., $\log Pr(\hat{y_i} = 1|\boldsymbol{x_i}, \boldsymbol{e_i}) > -\lambda^{(1)} - \alpha v_i^{(0)}$, are assigned with $v_i^{(1)} = 1$; otherwise, $v_i^{(0)} = 0$.

When we update the $\boldsymbol{v^{(0)}}$, the similar closed-form solution can be derived as follows.

$$v_i^{(0)} = \begin{cases} 1 & - \log \sigma(\gamma \boldsymbol{w_c}^T \boldsymbol{e_i}) < \lambda^{(0)} + \alpha v_i^{(1)} \\ 0 & \text{Otherwise} \end{cases} \quad (7)$$

The goal of this step is to formally define which graph context pairs (i, c, γ) will be fed into the training pool for learning the network embedding $\boldsymbol{E}$. In each iteration, the graph context pairs (i, c, γ) whose prediction losses are smaller than a certain threshold, i.e., $-\log \sigma(\gamma \boldsymbol{w_c}^T \boldsymbol{e_i}) < \lambda^{(0)} + \alpha v_i^{(1)}$, are selected ($v_i^{(0)} = 1$) to be fed into the following RCE DNN.

Furthermore, the consensus regularizer $\mathcal{L}_{co}$ is imposed on the self-paced vectors $\boldsymbol{v^{(0)}}$ and $\boldsymbol{v^{(1)}}$ to ensure the selected graph context pairs (i, c, γ) are rare category oriented and within the user-defined level of learning difficulty. The constant α is used to balance the two learning principals, i.e., learning from rare category related graph context ($v_i^{(1)} = 1$) or learning graph context with less difficulty ($v_i^{(0)} = 1$). To be more specific, when α is larger, $\boldsymbol{v^{(0)}}$ will be closer to $\boldsymbol{v^{(1)}}$ such that more rare category related graph context will be selected to train RCE DNN; when α is smaller, $\boldsymbol{v^{(0)}}$ will select more vertices with 'easy' graph context.

RCE in the Scarcity of Labeled Minority Classes Examples: To learn the graph embedding that preserves the similarities among rare category examples while maximally separating these examples from the majority class examples, we follow the negative-sampling-based graph embedding models [10, 37], which minimize the cross entropy loss of predicting graph context pairs (i, c) to positive labels ($\gamma = 1$) or negative labels ($\gamma = -1$) as follows:

$$\min -E_{(i,c,\gamma)} \log \sigma(\gamma \boldsymbol{\theta_c}^T \boldsymbol{e_i})$$

where $\sigma(x)$ is the sigmoid function, i.e., $\sigma(x) = 1/(1 + e^{-x})$. Recently, [44] further developed a label informed graph embedding method that injects the label information into the sampled positive graph context pairs and demonstrated its effectiveness in the semi-supervised learning setting. However, in our problem setting, the above methods may fail due to the following reasons: (1) the learned embeddings (e.g., [10, 37, 44]) are not sensitive to the minority class examples since the sampled graph context pairs using the above methods may mostly come from the majority classes; (2) the scarcity of the labeled minority class examples imposes severe limitation on sampling rare category oriented graph context pairs. In the extreme case, when there is only one labeled minority example, the existing method [44] cannot generate the label informed positive context pairs $(i, c, +1)$ as there is no way to find a pair of nodes (i, c) from the same minority class within the labeled set.

To address the above deficiencies, we develop a rare category oriented context sampling strategy in Algorithm 1. The given input of Algorithm 1 is the graph G, an indicator vector $\boldsymbol{I}$, and some constant parameters including the length of the performed random walks μ, the probability r and the number of negative samples s_{neg}. In particular, the indicator vector $\boldsymbol{I}$ can be generated by any offline RCC models, while our proposed *SPARC* model utilizes the self-paced vector $\boldsymbol{v^{(1)}}$ to serve as the indicator vector $\boldsymbol{I}$ that determines

Algorithm 1 Rare Category Oriented Context Sampling

```
Input:
    Graph $G$, indicator vector $\boldsymbol{I}$ and parameters $\mu$, $r$ and $s_{neg}$.
Output:
    Rare category oriented graph context pairs;
Draw a number $random \sim Unif(0, 1)$.
if $random < r$ then
    Uniformly sample a random walk $\mathbb{W}$ of length $\mu$ and generate
    one positive graph context pair $(i, c, +1)$ and $s_{neg}$ negative
    graph context pairs $(i, c, -1)$ by existing methods [31, 37].
else
    Shuffle an initial vertex $v_i$ from the nonzero elements in $\boldsymbol{I}$
    and conduct a random walk $\mathbb{W}$ of length $\mu$.
    Uniformly sample a positive graph context pair $(i, c, +1)$ with
    $I(i) = I(c)$ and $s_{neg}$ negative graph context pairs $(i, c, -1)$
    with $I(i) \neq I(c)$.
end if
```

the potential rare category examples based on the current RCC DNN. Algorithm 1 samples two types of graph contexts, i.e., the general graph context and the rare category related graph context, where the first one preserves the general graph structure, while the second one focuses on learning the local context of the rare category examples. An example of sampling graph context is shown in Fig. 3. With probability r, the general graph contexts are extracted by the existing methods [31, 37]. With probability $(1 - r)$, we sample rare category related context pairs (i, c, γ). In particular, when $\gamma = +1$, node i and node c are believed to belong to the same minority class, i.e., $I(i) = I(c)$; when $\gamma = -1$, node i and node c are believed to belong to the different classes, i.e., $I(i) \neq I(c)$.

Remarks: We would like to emphasize that Algorithm 1 is designed for the class-imbalanced networks. More specifically, (1) to counter the skewed distribution when sampling graph pairs, Algorithm 1 uses a probability r to balance the proportion of general graph context pairs and the rare category graph context pairs; (2) in scarcity of labeled rare category examples, our method generates rare category oriented graph context pairs (i, c, γ) based on the pseudo labels (i.e., indicator vector I) instead of using real labels to alleviate the limitation of insufficient labeled examples.

RCC with Respect to Labeled Majority and Minority Class Examples: Here, we show the underlying training process of the RCC DNN regarding the labeled majority class examples and the labeled minority class examples. For each labeled minority class example i, the hidden layers of RCC DNN are updated by minimizing the following objective.

$$\begin{aligned}\mathcal{L}_{min} &= c_{1,0} \log Pr(\hat{y}_i = 0|\boldsymbol{x_i}, \boldsymbol{e_i}) - v_i^{(1)} \log Pr(\hat{y}_i = 1|\boldsymbol{x_i}, \boldsymbol{e_i}) \\ &= c_{1,0} \log(1 - Pr(\hat{y}_i = 1|\boldsymbol{x_i}, \boldsymbol{e_i})) - v_i^{(1)} \log Pr(\hat{y}_i = 1|\boldsymbol{x_i}, \boldsymbol{e_i})\end{aligned} \quad (8)$$

To further simplify the above objective, we let $a = 1 - Pr(\hat{y}_i = 1|\boldsymbol{x_i}, \boldsymbol{e_i})$ and $b = Pr(\hat{y}_i = 1|\boldsymbol{x_i}, \boldsymbol{e_i})$. Since $(a + b)^2 \geq 4ab$, we have $2 \log(a + b) \geq \log 4 + \log a + \log b$, which could be written in terms of $Pr(\hat{y}_i = 1|\boldsymbol{x_i}, \boldsymbol{e_i})$ as follows:

$$\log(1 - Pr(\hat{y}_i = 1|\boldsymbol{x_i}, \boldsymbol{e_i})) \leq -\log Pr(\hat{y}_i = 1|\boldsymbol{x_i}, \boldsymbol{e_i}) - \log 4 \quad (9)$$

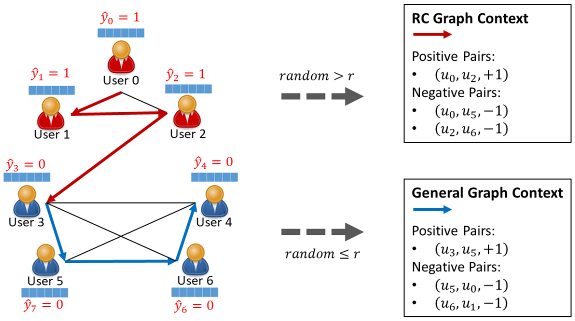


Figure 3: An example of sampling dual graph context by Algorithm 1, when the window size $d = 2$, $\mu = 3$ and $s_{neg} = 2$. In particular, if $random > r$, we sample the rare category related context pairs based on the random walk (e.g., red path) starting from the labeled rare category example (e.g., User0); otherwise, we extract the graph context by uniformly sampling a random walk (e.g., blue path) from the given network.

By substituting Eq. 9 back into Eq. 8, we have:

$$\mathcal{L}_{min} \leq -(c_{1,0} + v_i^{(1)}) \log Pr(\hat{y}_i = 1|\boldsymbol{x_i}, \boldsymbol{e_i}) - c_{1,0} \log 4$$

Similar as above, for each labeled majority class example j, the RCC DNN aims to minimize the following objective:

$$\mathcal{L}_{maj} = (c_{0,1} - v_j^{(1)}) \log Pr(\hat{y}_j = 1|\boldsymbol{x_j}, \boldsymbol{e_j}) \quad (10)$$

Remarks: Based on the above derived objectives regarding labeled majority class examples and labeled minority class examples, we have the following observations: (1) $\mathcal{L}_{min}$ is monotonically decreasing over $Pr(\hat{y}_i = 1|\boldsymbol{x}_i, \boldsymbol{e}_i))$ as $c_{1,0} > 1$ and $v_i^{(1)} \in \{0, 1\}$. That is, the probability of the labeled minority class examples ($y = 1$) belonging to the minority class $Pr(\hat{y}_i = 1|\boldsymbol{x}_i, \boldsymbol{e}_i)$ is maximized along with minimizing $\mathcal{L}_{min}$. (2) $\mathcal{L}_{maj}$ is monotonically nondecreasing over $Pr(\hat{y}_j = 1|\boldsymbol{x}_j, \boldsymbol{e}_j))$ as $c_{0,1} - v_i^{(1)} \geq 0$. That is, the probability of the labeled majority class examples ($y = 0$) belonging to the minority class $Pr(\hat{y}_i = 1|\boldsymbol{x}_j, \boldsymbol{e}_j)$ is minimized along with optimizing $\mathcal{L}_{min}$. (3) The overall objective of *SPARC* emphasizes on learning the underlying distribution of the minority class as $c_{1,0} + v_i^{(1)} > c_{0,1} - v_j^{(1)}$. For a special case, when $c_{0,1} - v_j^{(1)} = 0$, i.e., $c_{0,1} = v_j^{(1)} = 1$, the labeled majority class examples with $v_j^{(1)} = 1$ are not taken into consideration. The intuition is that our proposed framework *SPARC* is designed to be tolerant of the majority class example j that may not be separable from the minority class examples, i.e., $\log Pr(\hat{y}_j = 1|\boldsymbol{x}_j, \boldsymbol{e}_j) > -\lambda^{(1)} - \alpha v_i^{(0)}$.

4.2 Optimization Algorithm

To optimize the overall objective function in Eq. 2, we adopt stochastic gradient descent (SGD) [5] to train our model in an alternative way. The optimization algorithm is summarized in Algorithm 2. The given input is the attributed network G, labels of training data $Y = \{y_1, \ldots, y_L\}$ and some parameters including batch iterations T_1 and T_2, batch size N_1 and N_2, self-paced parameters $\lambda^{(0)}$ and $\lambda^{(1)}$ and α. Within each iteration, we first sample

Algorithm 2 *SPARC*: Joint Learning Framework for RCC and RCE

```
Input:
    Graph $G = (V, E, X)$, labels $Y = \{y_1, \ldots, y_L\}$, and parameters
    $T_1, T_2, N_1, N_2, \lambda^{(0)}, \lambda^{(1)}, \alpha$.
Output:
    (1) Rare category oriented embedding $E \in \mathcal{R}^{|V| \times d}$;
    (2) A list of predicted rare category examples.
 1: while Stopping criterion is not satisfied do
 2:     for $t = 1 : T_1$ do
 3:         Sample $N_1$ labeled instances and update hidden layers'
            parameters $\theta$ by taking a gradient step for $\mathcal{L}_s + \mathcal{L}_{rc}$.
 4:     end for
 5:     for $t = 1 : T_2$ do
 6:         Sample $N_2$ graph context pairs by Algorithm 1 with indicator vector $\boldsymbol{v}^{(0)}$.
 7:         Update the rare category oriented embedding $E$ by taking
            a gradient step for $\mathcal{L}_{tc}$
 8:     end for
 9:     Update $\boldsymbol{v}^{(0)}$ and $\boldsymbol{v}^{(1)}$ separately based on Eq. 7 and Eq. 6,
        and make sure all the labeled rare examples are selected.
10:     Augment $\lambda^{(0)}, \lambda^{(1)}$.
11: end while
```

N_1 labeled examples and update RCC DNN by taking a gradient step of $\mathcal{L}_s + \mathcal{L}_{rc}$. Note that, in the first iteration, $\mathcal{L}_{rc} = 0$ as $\boldsymbol{v}^{(0)}$ and $\boldsymbol{v}^{(1)}$ are initialized to all-zero vectors. We then optimize the RCE DNN over N_2 sampled graph context pairs (i, c, γ). The above procedures are repeated with T_1 and T_2 times respectively. Step 9 updates the self-paced vectors $\boldsymbol{v}^{(0)}$ and $\boldsymbol{v}^{(1)}$, and Step 10 augments the self-paced parameters $\lambda^{(0)}, \lambda^{(1)}$ in order to learn the more 'difficult' concept in the next iterations. The algorithm stops when the user-defined stopping criterions are satisfied.

Algorithm 2 can be extended to solve the multi-class RCE and RCC problems by optimizing the following objective function.

$$\begin{aligned}\mathcal{L}_m = & \sum_{i=1}^{L}\sum_{c=1}^{C} c_{y_i,\hat{y}_i} \log Pr(\hat{y_i} = c|\boldsymbol{x}_i, \boldsymbol{e}_i) \\ & - \sum_{i=1}^{L+U}\sum_{c=1}^{C} v_i^{(c)} \log Pr(\hat{y_i} = c|\boldsymbol{x_i}, \boldsymbol{e_i}) + v_i^{(0)} \log E_{(i,c,\gamma)} \log \sigma(\gamma \boldsymbol{\theta_c}^T \boldsymbol{e}_i) \\ & - \sum_{i=1}^{L+U}[\sum_{c=1}^{C} \lambda^{(c)} v_i^{(c)} + \lambda^{(0)} v_i^{(0)}] - \alpha \sum_{i=1}^{L+U}\sum_{c=1}^{C} v_i^{(c)} v_i^{(0)} \qquad (11)\end{aligned}$$

where $\boldsymbol{v}^{(c)} \in [0, 1]^n$ denotes the self-paced vector of class c, and $\lambda^{(c)}$ is the self-paced parameter that controls the learning pace.

Compared with the objective function in Eq. 2 for the binary case, the only difference is that each term in Eq. 11 is defined by cumulating the prediction loss over multiple classes instead of only two. Following Algorithm 2, our proposed framework *SPARC* is iteratively trained based on the extracted graph context pairs and label propagated examples that come from different classes. In the end, *SPARC* returns the rare category oriented network representation and the predication labels of each vertex in the given network.

5 EXPERIMENTAL RESULTS

In this section, we demonstrate the performance of our proposed *SPARC* algorithm in the sense of the saliency of the RCE representation and the accuracy of the RCC classifier for rare category analysis. Moreover, we also present a case study to illustrate the impacts and the underlying procedures of the self-paced learning in our proposed *SPARC* framework.

5.1 Experiment Setup

Category	Network	Classes	Smallest Class	Nodes	Edges
Collaboration	DBLP	20	1.91%	2,309	7,913
	SO	2	1.29%	3,262	19,926
NLP	Citeseer	6	3.42%	3,327	4,732
	Cora	7	1.14%	2,708	5,429
	Pubmed	3	4.05%	19,717	44,318
Social	Epinion	19	1.38%	75,879	508,837

Table 1: Statistics of the network data sets.

Data sets: The statistics of all real data sets used in our experiments are summarized in Table 1.

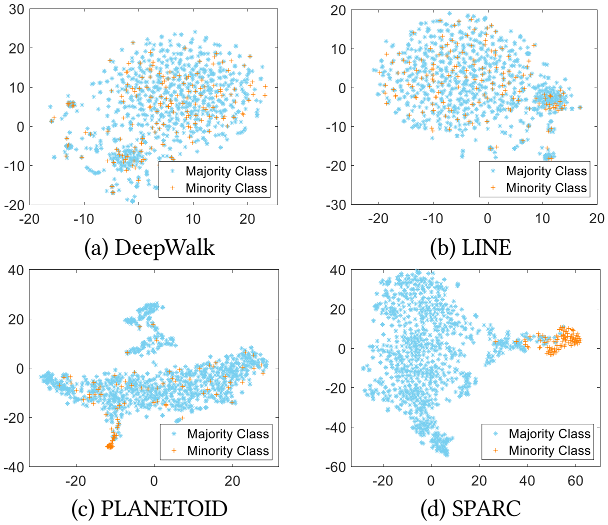
- **Collaboration Networks:** DBLP[*] data set provides the bibliographic information of the publications in IEEE Visualization Conference during 1990 ~ 2015. Each vertex represents a paper, and an edge exists if and only if when one paper cites another paper. The class membership is defined based on 20 research topics in the data visualization area. SO[†] data set is collected from Stack Overflow, where each node represents a Stack Overflow user and each edge indicates one comment from one user to another. The class memberships are defined based on the users' reputation score, i.e., the majority of the users have regular scores (< 3000) while only a few users have considerably high scores (> 3000).
- **NLP Networks:** Citeseer, Cora and Pubmed are three text classification data sets[‡], where each node represents a document and each edge indicates the citation link between the documents. The bag-of-words representation is adopted as the node attributes in these three data sets. NELL [44] is an entity classification data set, where the entities and the relations between entities are extracted from the NELL knowledge database, and the attributes of each entity are obtained by the bag-of-words representation of the associated description text.
- **Social Network:** Epinion [22] data set is a who-trust-whom social network, where each node represents a user, and an edge exits if and only if two users both give positive reviews (rating > 2.5 out of 5) to the same item. The class membership of each user is defined based on the most frequently reviewed item category.

Comparison Methods: We compare *SPARC* with the recent network embedding and rare category analysis models. DeepWalk [31] and LINE [37] are unsupervised network embedding algorithms, which learn embedding based on word2vec model and use logistic

[*] http://www.vispubdata.org/site/vispubdata/

[†] https://archive.org/details/stackexchange

[‡] http://linqs.umiacs.umd.edu/projects//projects/lbc/



(a) DeepWalk (b) LINE

(c) PLANETOID (d) SPARC

Figure 4: 2-D t-SNE visualization of network embedding.

regression as the classifier. PLANETOID [44] is a semi-supervised framework for attributed networks, which learns an embedding based on both topology context and label context to better infer the class memberships of unlabeled examples. GRADE [13] is a graph based rare category detection algorithm that takes the input of adjacency matrix $\boldsymbol{A}$, while RACH [14] is a rare category characterization algorithm that takes the feature vectors $\boldsymbol{X}$ as input.

Repeatability: All the data sets are publicly available. We will release the code of our algorithms through the authors' website after the paper is published. The experiments are performed on a Windows machine with four 3.5GHz Intel Cores and 256GB RAM.

5.2 Network Layout

A simple but useful way to evaluate the network representation approaches is to visualize the network layout in the embedding space, and we take the NLP network that extracted from the Pubmed data set for an example. We separate the network into binary classes by letting the smallest class be the minority class and the residual be the majority class. Laying out this NLP network is very challenging as the data is noisy and the classes, i.e., categories of documents, always overlap with one another. We compare our proposed *SPARC* algorithm with three state-of-the-art network embedding algorithms including two unsupervised methods, i.e., DeepWalk and LINE, and one semi-supervised method, i.e., PLANETOID. Note that, the unsupervised embedding methods only take as input the graph G, while the semi-supervised methods, i.e., PLANETOID and our proposed *SPARC*, are further provided with the training data consisting of labeled examples from both the majority and the minority classes. In particular, we first map the given network into a 129-dimensional space with different embedding methods, and then we employ the nonlinear dimensionality reduction method, i.e., t-SNE [28], to a 2-D space for the better visualization, which is shown in Fig. 4. We can clearly observe that (1) the semi-supervised embedding methods perform better than the unsupervised methods as the classes are better separated; (2) with the same amount of training data, the rare examples are better clustered by using *SPARC* than PLANETOID. One explanation is that PLANETOID samples the graph context without considering that the class membership is imbalanced, which results in the neighborhood context of rare examples not well preserved in the embedding space.

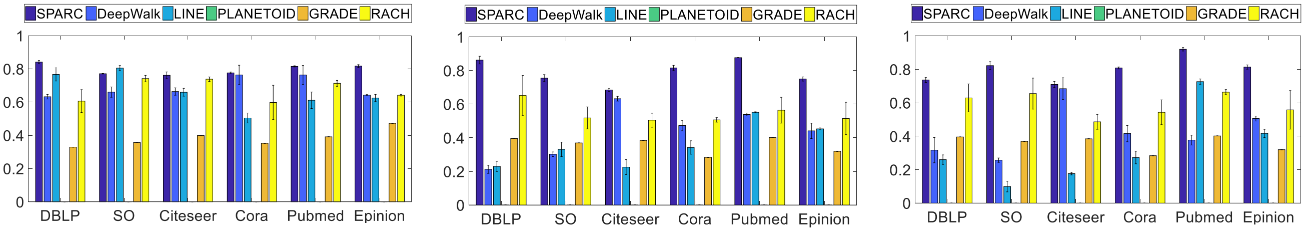
5.3 Effectiveness Analysis

The comparison results in terms of effectiveness across a diverse set of networks by using 1, 5% and 10% labeled rare category examples are shown from Fig. 5 to Fig. 7, where the height of the bars indicates the average value of evaluation metrics, and the error bars represent the standard deviation of evaluation metrics in multiple runs by randomly shuffling the initial training examples. Note that PLANETOID can not be trained with only one labeled rare category example, thus the corresponding results are not reported in Fig. 5. By considering the smallest class in each data set as the rare category, we adopt the following three commonly used metrics for the rare category analysis [14]: (1) accuracy, which measures the rate of the correctly classified majority and minority class examples; (2) recall, which measures the percentage of the discovered rare category examples; (3) recall@K, which shows the ratio of true rare examples being retrieved in the returned top K examples, where K equals the number of rare category examples in the given network. In general, we observe that: (1) Our proposed *SPARC* algorithm outperforms the comparison methods across all the data sets and evaluation metrics in most cases. For example, on DBLP network with only one labeled minority class example, compared with the best competitor RACH, *SPARC* is 39% higher on Accuracy, 32% higher on Recall and 17% higher on Recall@K. (2) Our proposed *SPARC* algorithm is more robust (i.e., smaller error bar) than the comparison methods with different initial training examples. One intuitive explanation might be that the training 'curriculum' generated by *SPARC* guides the learning process towards a better local optimum in the parameter space.

5.4 Case Study: Impact of Self-Paced Learning

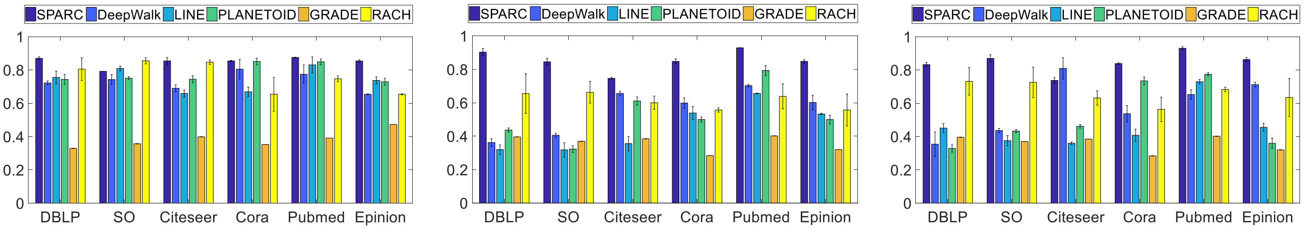
Dual Graph Context Selection: To illustrate the impact of SPL on RCE, we conduct a case study on Pubmed to show how the rare category oriented graph contexts are extracted over paces. In particular, we show the vertices that were selected by the self-paced vectors $\boldsymbol{v}^{(0)}$ and $\boldsymbol{v}^{(1)}$ on the final embedding space of *SPARC*. Remember the self-paced vectors are updated over iterations (i.e., paces) by shifting from the 'easy' concept to the target 'difficult' one. In Fig. 8, we observe that: (1) In the initial iteration (i.e., Pace 0), no vertices are selected by $\boldsymbol{v}^{(0)}$ and $\boldsymbol{v}^{(1)}$. (2) After that, $\boldsymbol{v}^{(1)}$ mainly selects the examples in the region of the minority class, while $\boldsymbol{v}^{(1)}$ selects examples across the whole network. (3) From Pace 1 to Pace 9, the overlap between the selected examples in $\boldsymbol{v}^{(0)}$ and $\boldsymbol{v}^{(1)}$ is increasing, which indicates that the RCE emphasizes on learning the context information of the minority class.

Parameter Sensitivity: We study the sensitivity of self-paced parameters $\lambda^{(0)}$ and $\lambda^{(1)}$ on Pubmed. Recall that $\lambda^{(0)}$ and $\lambda^{(1)}$ control the paces of learning from graph context and the underlying distribution of rare examples. In Fig. 9, we report the recall rates of *SPARC* by iteratively augmenting the values of $\lambda^{(0)}$ and $\lambda^{(1)}$. We have the following observations: (1) Recall is generally increasing with the



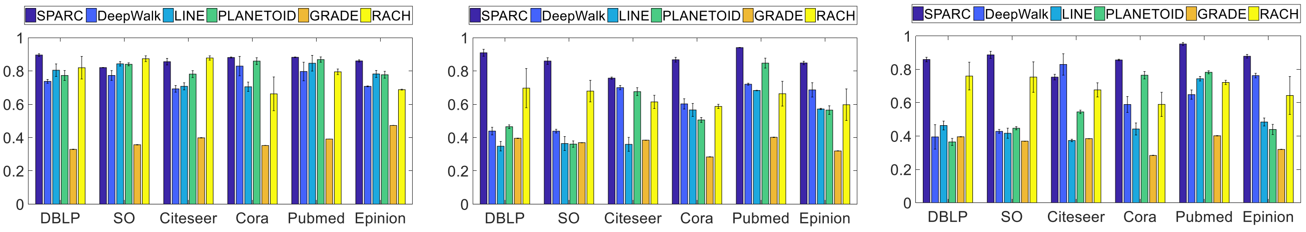
(a) Accuracy (b) Recall (c) Recall@K

Figure 5: Effectiveness analysis with one labeled minority class example.



(a) Accuracy (b) Recall (c) Recall@K

Figure 6: Effectiveness analysis with 5% labeled minority class examples.



(a) Accuracy (b) Recall (c) Recall@K

Figure 7: Effectiveness analysis with 10% labeled minority class examples.

values of $\lambda^{(0)}$ and $\lambda^{(1)}$ over paces. An intuitive explanation is that when $\lambda^{(0)}$ and $\lambda^{(1)}$ are augmented, the richer context information of rare examples is extracted for training *SPARC* according to Eq. 6 and Eq. 7, which leads to a better prediction model. (2) In the early stage (i.e., $\lambda^{(0)} = \lambda^{(1)} = 0.16$), recall increases faster (slower) with respect to $\lambda^{(1)}$ ($\lambda^{(0)}$). In other words, learning from rare examples with propagated labels is more important than learning from the graph context for the RCC task in the initial iterations.

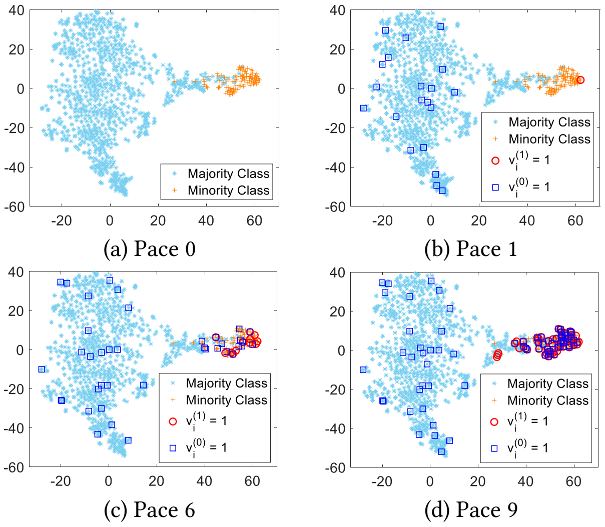
6 CONCLUSION

In this paper, we focus on analyzing the rare categories in class-imbalanced networks. We start by formally defining the RCE and RCC problems related to the rare categories, and then identify their unique challenges due to the nature of rare categories in the attributed networks, i.e., highly skewness, non-separability and label scarcity. To address these challenges, we propose a generic rare category analysis framework named *SPARC*, which jointly learns the network representation and rare category characterization model in a mutually beneficial way by shifting from the 'easy' concept to the target 'difficult' one, in order to facilitate more reliable label propagation to the large number of unlabeled examples. The empirical evaluations on real-world data sets demonstrate the effectiveness of our proposed framework *SPARC* from multiple perspectives.

ACKNOWLEDGMENT

This work is supported by the United States Air Force and DARPA under contract number FA8750-17-C-0153 [§], National Science Foundation under Grant No. IIS-1552654, Grant No. IIS-1813464 and Grant No. CNS-1629888, the U.S. Department of Homeland Security under Grant Award Number 2017-ST-061-QA0001, and an IBM Faculty Award. The views and conclusions are those of the authors and should not be interpreted as representing the official policies of the funding agencies or the government.

[§] Distribution Statement "A" (Approved for Public Release, Distribution Unlimited)



(a) Pace 0

(b) Pace 1

(c) Pace 6

(d) Pace 9

Figure 8: Self-paced dual graph context selection.

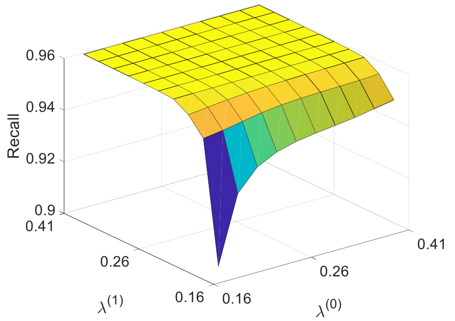


Figure 9: Parameter sensitivity analysis.

REFERENCES

[1] M. Belkin and P. Niyogi. 2002. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS (2002)*.

[2] Y. Bengio. 2014. Evolving culture versus local minima. In *Growing Adaptive Machines (2014)*.

[3] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. 2009. Curriculum learning. In *ICML (2009)*.

[4] A. Blum and T. Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *COLT (1998)*.

[5] L. Bottou. 2010. Large-scale machine learning with stochastic gradient descent. In *COMPSTAT (2010)*.

[6] N. V Chawla, K. W Bowyer, L. O Hall, and W P. Kegelmeyer. 2002. SMOTE: synthetic minority over-sampling technique. *JAIR* (2002).

[7] T. Chen and Y. Sun. 2017. Task-Guided and Path-Augmented Heterogeneous Network Embedding for Author Identification. In *ACM WSDM (2017)*.

[8] B. Du, S. Zhang, N. Cao, and H. Tong. 2017. First: Fast interactive attributed subgraph matching. In *ACM SIGKDD (2017)*.

[9] E. M Fich and A. Shivdasani. 2007. Financial fraud, director reputation, and shareholder wealth. *Journal of Financial Economics* (2007).

[10] A. Grover and J. Leskovec. 2016. node2vec: Scalable feature learning for networks. In *ACM SIGKDD (2016)*.

[11] R. Guo, J. Li, and H. Liu. 2018. INITIATOR: Noise-contrastive Estimation for Marked Temporal Point Process. In *IJCAI (2018)*.

[12] H. He and E. A Garcia. 2009. Learning from imbalanced data. *IEEE TKDE* (2009).

[13] J. He, Y. Liu, and R. Lawrence. 2008. Graph-based rare category detection. In *IEEE ICDM (2008)*.

[14] J. He, H. Tong, and J. Carbonell. 2010. Rare category characterization. In *IEEE ICDM (2010)*.

[15] V. Hodge and J. Austin. 2004. A survey of outlier detection methodologies. *Artificial intelligence review* (2004).

[16] C. Huang, Y. Li, C. Change, and X. Tang. 2016. Learning deep representation for imbalanced classification. In *IEEE CVPR (2016)*.

[17] L. Jiang, D. Meng, Q. Zhao, S. Shan, and A. G Hauptmann. 2015. Self-Paced Curriculum Learning.. In *AAAI (2015)*.

[18] N Jindal and B Liu. 2007. Review spam detection. In *WWW (2007)*.

[19] F. Khan, B. Mutlu, and X. Zhu. 2011. How do humans teach: On curriculum learning and teaching dimension. In *NIPS (2011)*.

[20] J. B Kruskal and M. Wish. 1978. *Multidimensional scaling*.

[21] M P. Kumar, B. Packer, and D. Koller. 2010. Self-paced learning for latent variable models. In *NIPS (2010)*.

[22] J. Leskovec and A. Krevl. 2015. {SNAP Datasets}:{Stanford} Large Network Dataset Collection. (2015).

[23] J. Li, C. Chen, H. Tong, and H. Liu. 2018. Multi-layered network embedding. *SIAM SDM* (2018).

[24] J. Li, H. Dani, X. Hu, J. Tang, Y. Chang, and H. Liu. 2017. Attributed network embedding for learning in a dynamic environment. In *ACM CIKM (2017)*.

[25] S. Li, M. Shao, and Y. Fu. [n. d.]. Multi-view low-rank analysis for outlier detection. In *SIAM SDM (2015)*.

[26] S. Li, M. Shao, and Y. Fu. 2018. Multi-View Low-Rank Analysis with Applications to Outlier Detection. *TKDD* (2018).

[27] F. Ma, D. Meng, Q. Xie, Z. Li, and X. Dong. 2017. Self-paced co-training. In *ICML (2017)*.

[28] L. Maaten and G. Hinton. 2008. Visualizing data using t-SNE. *JMLR* (2008).

[29] T. Mikolov, I. Sutskever, K. Chen, G. S Corrado, and J. Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS (2013)*.

[30] D. Pelleg and A. W Moore. 2005. Active learning for anomaly and rare-category detection. In *NIPS (2005)*.

[31] B. Perozzi, R. Al-Rfou, and S. Skiena. 2014. Deepwalk: Online learning of social representations. In *ACM SIGKDD (2014)*.

[32] X. R and L. Bo. 2012. Discriminatively trained sparse code gradients for contour detection. In *NIPS (2012)*.

[33] S. T Roweis and L. K Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *science* (2000).

[34] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang. 2015. Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection. In *IEEE CVPR (2015)*.

[35] V. I Spitkovsky, H. Alshawi, and D. Jurafsky. 2009. Baby Steps: How "Less is More" in unsupervised dependency parsing. *NIPS (2009)*.

[36] Y. Sun, M. S Kamel, and Y. Wang. 2006. Boosting for learning multiple classes with imbalanced class distribution. In *IEEE ICDM (2006)*.

[37] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei. 2015. Line: Large-scale information network embedding. In *WWW (2015)*.

[38] J. B Tenenbaum, V. De Silva, and J. C Langford. 2000. A global geometric framework for nonlinear dimensionality reduction. *science* (2000).

[39] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang. 2017. Community Preserving Network Embedding.. In *AAAI (2017)*.

[40] G. Wu and E. Y Chang. 2003. Adaptive feature-space conformal transformation for imbalanced-data learning. In *ICML (2003)*.

[41] J. Wu, J. He, and Y. Liu. 2018. ImVerde: Vertex-Diminished Random Walk for Learning Network Representation from Imbalanced Data. *arXiv preprint arXiv:1804.09222* (2018).

[42] S. Wu, Q. Ji, S. Wang, H. Wong, Z. Yu, and Y. Xu. 2017. Semi-Supervised Image Classification with Self-Paced Cross-Task Networks. *TMM* (2017).

[43] Y. Xu and W. Yin. 2013. A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *JIS* (2013).

[44] Z. Yang, W. W Cohen, and R. Salakhutdinov. 2016. Revisiting semi-supervised learning with graph embeddings. In *ICML (2016)*.

[45] B. Zadrozny, J. Langford, and N. Abe. 2003. Cost-sensitive learning by cost-proportionate example weighting. In *IEEE ICDM (2003)*.

[46] S. Zhang, D. Zhou, M. Y. Yildirim, S. Alcorn, J. He, H. Davulcu, and H. Tong. 2017. HiDDen: hierarchical dense subgraph detection with application to financial fraud detection. In *SIAM SDM (2017)*.

[47] D. Zhou, J. He, K S. Candan, and H. Davulcu. 2015. MUVIR: Multi-View Rare Category Detection. In *IJCAI (2015)*.

[48] D. Zhou, J. He, Y. Cao, and J. Seo. 2016. Bi-level rare temporal pattern detection. In *IEEE ICDM (2016)*.

[49] D. Zhou, A. Karthikeyan, K. Wang, N. Cao, and J. He. 2017. Discovering rare categories from graph streams. *DMKD* (2017).

[50] D. Zhou, K. Wang, N. Cao, and J. He. 2015. Rare category detection on time-evolving graphs. In *IEEE ICDM (2015)*.

[51] D. Zhou, S. Zhang, M. Y. Yildirim, S. Alcorn, H. Tong, H. Davulcu, and J. He. 2017. A local algorithm for structure-preserving graph cut. In *ACM SIGKDD (2017)*.