Active Discriminative Network Representation Learning

Li Gao*‡, Hong Yang†, Chuan Zhou*‡, Jia Wu♯, Shirui Pan†, Yue Hu*‡
*Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
†Centre for Artificial Intelligence, University of Technology Sydney, Australia
♯Department of Computing, Macquarie University, Sydney, Australia
‡School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
{gaoli, zhouchuan}@iie.ac.cn, Jia.Wu@mq.edu.au, {hong.yang@student., shirui.pan@}uts.edu.au

Abstract

Most of current network representation models are learned in unsupervised fashions, which usually lack the capability of discrimination when applied to network analysis tasks, such as node classification. It is worth noting that label information is valuable for learning the discriminative network representations. However, labels of all training nodes are always difficult or expensive to obtain and manually labeling all nodes for training is inapplicable. Different sets of labeled nodes for model learning lead to different network representation results. In this paper, we propose a novel method, termed as ANRMAB, to learn the active discriminative network representations with a multi-armed bandit mechanism in active learning setting. Specifically, based on the networking data and the learned network representations, we design three active learning query strategies. By deriving an effective reward scheme that is closely related to the estimated performance measure of interest, ANRMAB uses a multi-armed bandit mechanism for adaptive decision making to select the most informative nodes for labeling. The updated labeled nodes are then used for further discriminative network representation learning. Experiments are conducted on three public data sets to verify the effectiveness of ANRMAB.

1 Introduction

Network representation plays a critical role in the network analysis area [Perozzi *et al.*, 2014; Cavallari *et al.*, 2017], which aims to learn the embedding feature for each node, *i.e.,* embedding the nodes into a low-dimensional feature space while preserving their neighborhood relationships. To date, network representation learning has been successfully applied to many noteworthy applications, such as node classification [Tu *et al.*, 2016], recommendations [Gao *et al.*, 2016; Gao *et al.*, 2017] and link prediction [Cai *et al.*, 2018].

Most of previous network representation models are learned in unsupervised manner, and the learned representations are often weak in particular prediction scenarios, such as node classification[Yang *et al.*, 2016]. In reality, the network usually contains additional label information that summarizes the features of nodes. For example, documents in the citation network Citeseer are associated with field labels for the clear summarization and easy retrieval. It is worth pointing out that the label information is valuable for learning the discriminative network representations.

However, the labels of all training nodes are difficult, time-consuming and expensive to obtain, and manually labeling all the nodes for training is inapplicable. Different sets of labeled nodes for training lead to different network representation results [Cai *et al.*, 2017]. Thus, it is nontrivial to select nodes for labeling so as to maximize the performance of discriminative network representations. Active learning (AL) is demonstrated to be promising in solving the labeling problem [Zhu *et al.*, 2003; Aggarwal *et al.*, 2014; Konyushkova *et al.*, 2017]. The idea behind AL is that better models can be learned with less effort and lower cost by selecting training data cleverly, rather than at independent and identically distributed (i.i.d.) random [Zhang *et al.*, 2017].

Motivated by the above observations, with a given labeling budget, we aim to design a network representation learning method under the active learning principle to improve the performance by actively selecting the training labeled nodes. The objective mainly has the following challenges: 1) How to design the AL strategy based on the networking data, due to the fact that the nodes in the network are not independently and identically distributed; 2) With the given labeling budget and the strategy, how to adaptively select the informative nodes to label at each iteration; 3) The network representation learning and active learning affect each other, and how to jointly consider the both to maximize the performance.

Currently, there are few studies fully addressing the above challenges. The work [Cai *et al.*, 2017] proposes the method AGE that utilizes three AL query strategies to select the corresponding informative nodes to label. Different criteria are combined linearly using the time-sensitive parameters that follow the same distribution for each data set. However, considering that no single combinational module is likely to satisfy the characteristics of each data set, they fail to adaptively choose the nodes based on the estimated contributions to the

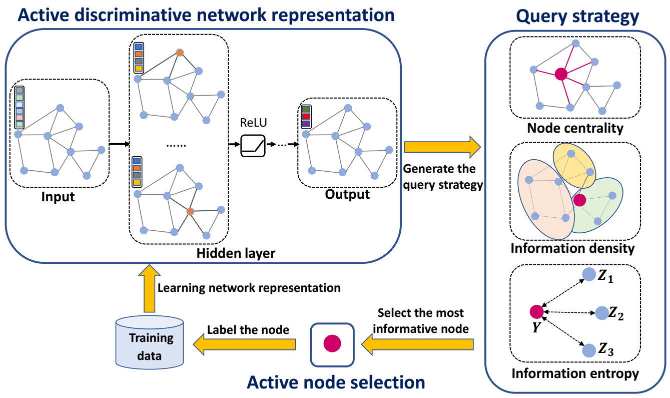


Figure 1: Illustration of the proposed method ANRMAB, which mainly contains three parts: query strategy component, active node selection component and active discriminative network representation component. In the query strategy component, the bigger red node in each AL strategy represents the corresponding informative node. Based on our designed multi-armed bandit mechanism, active node selection component gives the most informative nodes to label. The updated training data are then fed to the active discriminative network representation component, where the bars denotes the features or the activations of the nodes.

Notations	Descriptions
$\mathcal{G} = (\mathcal{V}, \mathcal{E})$	network $\mathcal{G}$ with nodes set $\mathcal{V}$ and edges set $\mathcal{E}$
$\mathcal{L}, \mathcal{U}$	the set of labeled ($\mathcal{L}$) and unlabeled ($\mathcal{U}$) nodes
B, M	labeling budget, feature dimensionality of a node
L_{init}, L_{max}	the initial and maximum amount of labeled nodes for training, $L_{max} = B + L_{init}$
C	the number of classes nodes belongs to in $\mathcal{G}$
N, K	the number of nodes in $\mathcal{G}$ and bandit machines
A, **F**	the adjacency matrix, the node feature matrix
Y	indicator matrix of the nodes having the labels
Z	output of the network representation model
R	the discriminative network representations
$\phi(v_i; \Theta)$	active learning query strategy ϕ for node v_i in $\mathcal{U}$ conditioned on parameters set Θ

Table 1: Notations and descriptions.

performance measure of interest. Thus, the learned network representations may be not promising for the classification task.

In this paper, we propose a novel method, termed as ANRMAB, to learn the **A**ctive discriminative **N**etwork **R**epresentations with a **M**ulti-**A**rmed **B**andit mechanism in the active learning setting, as illustrated in Figure 1. ANRMAB mainly contains three parts: query strategy component, active node selection component and active discriminative network representation component. In the query strategy component, three query strategies are designed based on the networking data and the trained network representations. Specifically, considering that the nodes are not i.i.d., but are associated with link relations, the node centrality that reflects its sociological origin [Newman, 2010] is given to evaluate the node's representativeness. The information entropy and the information density, which are widely used in most AL methods, are calculated (Challenge #1).

For the active node selection component, we use the multi-armed bandit mechanism for adaptive decision making by treating each query strategy as one bandit machine and deriving an effective reward scheme that is closely related to the estimated performance measure of interest. Based on our careful design of node selection process, the most informative nodes can then be chosen to label in each iteration (Challenge #2). For the active discriminative network representation component, we use the graph convolutional network (GCN) [Kipf and Welling, 2016] as an example method to derive the discriminative network representations.

The AL query strategy is generated at the end of the discriminative network representation learning. As more informative labeled nodes are provided for model training, network representations generally become more discriminative and accurate. Meanwhile, as the information density and information entropy are derived using the better trained network representations, the query strategies yield more informative nodes that will be queried by the active node selection component (Challenge #3).

The main contributions are summarized as follows.

- We propose a novel method ANRMAB for the active discriminative network representation learning with a multi-armed bandit mechanism in the AL setting.
- We give three AL query strategies based on the networking data and the trained network representations. A multi-armed bandit algorithm is designed to adaptively select the most informative node to label. Furthermore, the three components of ANRMAB affect each other and are collaboratively learned for better discriminative network representations.
- We conduct extensive experiments on three public data sets to verify the effectiveness of ANRMAB. We also conduct the visualization using t-SNE to illustrate the discrimination of the learned network representations.

2 Preliminaries

In this section, we introduce the preliminaries and then give our problem statement. The notations used throughout the paper are summarized in Table 1. The matrices and the vectors are marked as the bold characters.

2.1 Active Learning

We consider a pool-based AL setting [Settles and Craven, 2008; Zhang *et al.*, 2017] in this paper, where there exists a small set of labeled data $\mathcal{L}$ and a large pool of unlabeled data $\mathcal{U}$. The idea behind is to strategically make a query to select instances from $\mathcal{U}$ to label, so as to maximize the performance of the classification task. The query strategy, such as uncertainty sampling, is usually used to score the candidate instances in the pool.

The key for the query strategy is to design an informative measure. Let x^* be the most informative instance given a query strategy $\phi(x_i; \Theta)$, which evaluates each instance x_i in $\mathcal{U}$ conditioned on the current set of parameters Θ. The instance selection protocol can be defined as follows

$$x^* = \text{argmax}_{x_i \in \mathcal{U}} \phi(x_i; \Theta) \tag{1}$$

2.2 Multi-Armed Bandit Problem

The multi-armed bandit problem is a well-known adaptive learning problem [Gittins *et al.*, 1989; Vermorel and Mohri, 2005], which simulates what a gambler would do in a casino. Given K bandit machines and a budget of T iterations, the gambler then sequentially decides which machine to pull in each iteration t ($t = 1, \ldots, T$). Once being pulled, the bandit machine gives a reward randomly from a machine-specific distribution unknown to the gambler [Hsu and Lin, 2015]. The goal of the gambler is to maximize his total rewards in the series of trials.

3 Problem Statement

For a network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ along with its adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ and node feature matrix $\mathbf{F} \in \mathbb{R}^{N \times M}$, the notation $\mathcal{V}$ denotes the nodes set, $\mathcal{E}$ denotes the edges set, N is the number of nodes in $\mathcal{G}$ and M is the dimensionality of the feature vector of each node. Considering that different sets of training labeled nodes result in different model learning results, we aim to learn the discriminative network representations $\mathbf{R} \in \mathbb{R}^{N \times D}$ in the active learning setting, where D is the dimensionality of the node embedding space. Given labeling budget B and the number of initial labeled nodes L_{init}, the key point is to design the AL query strategy $\phi(v_i; \Theta)$ to actively select the most informative node v^* from the unlabeled set $\mathcal{U}$ in each iteration. The node v^* is then added to the labeled set $\mathcal{L}$ for further network representation learning, so as to maximize the performance of the classification task.

4 Our Solution: ANRMAB

In this section, we introduce our solution ANRMAB for learning the discriminative network representations with a multi-armed bandit mechanism in the AL setting. We first detail its main three parts: AL query strategy component, active node selection component and active discriminative network representation component, and then give a summarization of ANRMAB.

4.1 AL Query Strategy

As suggested by the AL literatures [Settles, 2010; Aggarwal *et al.*, 2014], uncertainty and representativeness are usually served as the measures. uncertainty sampling queries the nodes about which the classification model is least certain how to label. However, the nodes with greater uncertainty may not be representative of the data, and may refer to the noisy nodes or the outliers [Cai *et al.*, 2017]. The representativeness-based strategies are often used to balance the informativeness of the node with its uncertainty properties.

In our AL query strategy component, we adopt information entropy as the uncertainty measure, node centrality and information density as the representativeness measures.

Information Entropy

Given network $\mathcal{G}$, the adjacency matrix $\mathbf{A}$, node feature matrix $\mathbf{F}$ and the labeled nodes set $\mathcal{L}$, the information entropy based query strategy $\phi_{IE}(v_i; \Theta_{IE})$ for a candidate node v_i in $\mathcal{U}$ is defined as follows

$$\phi_{IE}(v_i; \Theta_{IE}) = -\sum_{c=1}^{C} P_{ic}(v_i; \mathbf{A}, \mathbf{F}, \mathcal{L}) \log P_{ic}(v_i; \mathbf{A}, \mathbf{F}, \mathcal{L}) \tag{2}$$

where $P_{ic}(v_i; \mathbf{A}, \mathbf{F}, \mathcal{L})$ is the probability of the candidate node v_i belonging to class c predicted by the active discriminative network representation component, which is detailed in Section 4.3, and $\Theta_{IE} := \{\mathbf{A}, \mathbf{F}, \mathcal{L}\}$ is the parameters set. The larger $\phi_{IE}(v_i; \Theta_{IE})$ is, the more uncertain the current model is with respect to node v_i.

Node Centrality

Considering that nodes in the network are not i.i.d., but are associated with link relations, the node centrality based on the network structure is given to evaluate the node's representativeness. There have been various metrics evaluating the centrality of a node, such as degree centrality [Newman, 2010], closeness centrality [Stephenson and Zelen, 1989] and PageRank centrality [Page *et al.*, 1999]. We adopt PageRank centrality, which has better performance for learning [Cai *et al.*, 2017], to calculate the query strategy.

The node centrality based query strategy $\phi_{NC}(v_i; \Theta_{NC})$ for a candidate node v_i in $\mathcal{U}$ is defined as follows

$$\phi_{NC}(v_i; \Theta_{NC}) = \frac{1-d}{N} + d \sum_j \mathbf{A}_{ji} \frac{\phi_{NC}(v_j; \Theta_{NC})}{\sum_k \mathbf{A}_{jk}} \tag{3}$$

where d is the damping factor and $\Theta_{NC} := \{\mathbf{A}\}$ is the parameter set. The node with the larger node centrality is favored in the node selection process.

Information Density

With the network representations $\mathbf{R}$ learned by the active discriminative network representation component, we can find the representative node in the latent D-dimensional embedding space. We apply the simple Kmeans method on the embeddings of all the nodes in $\mathcal{U}$ to calculate the information density of each candidate node v_i. The information density based query strategy $\phi_{ID}(v_i; \Theta_{ID})$ for a candidate node v_i in $\mathcal{U}$ is calculated as follows

$$\phi_{ID}(v_i; \Theta_{ID}) = \frac{1}{1 + d(\mathbf{R}_{v_i}, c_{v_i})} \tag{4}$$

where $d(\cdot)$ is the distance measure, such as the Euclidean distance, in the embedding space, $\mathbf{R}_{v_i}$ is the node embedding of v_i, and c_{v_i} is the center of the cluster that v_i belongs to. $\Theta_{ID} := \{\mathbf{R}\}$ is the parameter set.

4.2 Active Node Selection

Given the labeling budget B, it is intuitive that no single strategy is likely to satisfy the needs of each data set and to find the most informative node in each iteration. Our active node selection component aims to use a multi-armed bandit mechanism to adaptively choose the most informative node by treating each AL query strategy as one bandit machine and deriving an effective reward scheme that is closely related to the estimated performance measure of interest.

Multi-Armed Bandit Method

As the learning performance generally becomes better when $\mathcal{L}$ becomes larger after adding the labeled node in each iteration t ($t = 1, \dots, B$), it is intuitive that the rewards, which are associated with the learning performance, are not independent random variables across the iterations. Meanwhile, the contributions to the learning performance can be time-sensitive as different query strategies may select different nodes in different iterations [Donmez *et al.*, 2007]. The above observations can satisfy the adversarial setting in the multi-armed bandit problem [Auer *et al.*, 2002].

To adaptively choose the most informative node from the designed query strategies, we adjust the method ALBL proposed in [Hsu and Lin, 2015], which modifies the EXP4.P method [Beygelzimer *et al.*, 2011] that is with a strong theoretical guarantee for the adversarial setting.

Let $\mathbf{w}^t = (\mathbf{w}_1^t, \dots, \mathbf{w}_K^t)$ be the adaptive weight vector in iteration t, where the k-th element $\mathbf{w}_k^t$ is the non-negative weight of the k-th AL query strategy. We then scale the weight vector $\mathbf{w}^t$ to a probability vector $\mathbf{p}^t = (\mathbf{p}_1^t, \dots, \mathbf{p}_K^t)$, where the value of the k-th element is $\mathbf{p}_k^t \in [p_{min}, 1]$ with some parameter $p_{min} > 0$. In practice, p_{min} is set as $p_{min} = \sqrt{\frac{\ln K}{KB}}$. Our method ANRMAB chooses the AL query strategy based on the probability vector $\mathbf{p}^t$ and obtains the corresponding reward of this action.

We introduce the query matrix $\mathbf{Q}^t \in \mathbb{R}^{K \times N_u^t}$ to make a probability query from $\mathcal{U}^t$, where N_u^t is the number of nodes in $\mathcal{U}^t$ in iteration t and the element $\mathbf{Q}_{kj}^t$ denotes the preference of the k-th AL query strategy on querying the label of $x_j \in \mathcal{U}^t$ in iteration t, $\sum_{j=1}^{N_u^t} \mathbf{Q}_{kj}^t = 1$.

To sum up, given the AL query strategies, ANRMAB takes the probabilistic decision to sample the node v_j from the unlabeled nodes set $\mathcal{U}^t$ in t-th iteration based on the probability ψ_j^t, which is defined as follows

$$\psi_j^t = \sum_{k=1}^{K} \mathbf{p}_k^t \mathbf{Q}_{kj}^t \tag{5}$$

Reward Scheme

Considering that the test accuracy, which is the targeted performance measure of interest, is usually not available because a test set is generally impossible to obtain for AL, we here adjust the importance weighting technique [Beygelzimer *et al.*, 2009; Ganti and Gray, 2012] to define the reward scheme. Assume that the node v_i is selected to query in iteration t, for the classifier f^t which is learned by our active discriminative network representation component, the reward function $r^t(v_i; f^t, \tau)$ is defined after τ iterations as follows

$$r^t(v_i; f^t, \tau) = \frac{1}{B} \sum_{t=1}^{\tau} \frac{\widetilde{W}^t}{N_u^t} I(y_j = f^t(v_j)) \tag{6}$$

where $\widetilde{W}^t = (\psi_j^t)^{-1}$, y_j is the label of the node v_j and $f^t(v_j)$ is the predicted label of v_j.

ANRMAB takes $r^t(v_i; f^t, \tau)$ as the reward in iteration t to evaluate how much the chosen query strategy helps improving the classifier f^t. Considering that different query strategies may suggest the same node, we use $\frac{r^t(v_i; f^t, \tau)\mathbf{Q}_{k*}^t}{\psi_*^t}$ to update $\mathbf{w}_k^t$ on all the query strategies that make the same suggestion for selecting the node v_*.

Algorithm 1: The proposed ANRMAB algorithm.

Input: network $\mathcal{G}$, adjacency matrix $\mathbf{A}$, node feature matrix $\mathbf{F}$, nodes sets $\{\mathcal{L}, \mathcal{U}\}$, budget $T = B$, iteration $t = 1$.
Output: the final network representation $\mathbf{R}^{T+1}$.

calculate $\widehat{\mathbf{A}} = \widetilde{\mathbf{D}}^{-\frac{1}{2}} \widetilde{\mathbf{A}} \widetilde{\mathbf{D}}^{-\frac{1}{2}}$, and set $\mathbf{w}_k^1 = 1$ for $k = 1, \dots, K$.
calculate query strategy ϕ_{NC}^t for $t = 1, \dots, T$ using Eq. (3).
while $t \leq T$ **do**
  obtain $\mathbf{R}^t$ and the classifier f^t from method in Section 4.3.
  calculate query strategies ϕ_{IE}^t and ϕ_{ID}^t using Eqs. (2), (4).
  set $W^t = \sum_{k=1}^{K} \mathbf{w}_k^t$, and for $k = 1, \dots, K$, set $\mathbf{p}_k^t = (1 - Kp_{min}) \sum_{j=1}^{K} \frac{\mathbf{w}_j^t}{W^t} + p_{min}$.
  calculate query matrix $\mathbf{Q}^t$ and probability ψ^t using Eq. (5).
  sample the informative node v^* based on ψ^t.
  update the nodes sets $\mathcal{L}^t$ and $\mathcal{U}^t$ using v^*.
  calculate the reward scheme $r^t(v_i; f^t, \tau)$ using Eq. (6).
  set $\widehat{r} = \frac{r^t(v_i; f^t, \tau)\mathbf{Q}_{k*}^t}{\psi_*^t}$, $t = t + 1$, update the weight vector $\mathbf{w}_k^{t+1}$ using $\mathbf{w}_k^{t+1} = \mathbf{w}_k^t e^{\left(\frac{p_{min}}{2}\left(\widehat{r}_k + \frac{1}{\mathbf{p}_k^t}\sqrt{\frac{\ln(N/0.1)}{KT}}\right)\right)}$.
return *the discriminative network representation* $\boldsymbol{R}^{T+1}$.

4.3 Active Discriminative Network Representation

Given labeled nodes set $\mathcal{L}$ and network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with its adjacency matrix $\mathbf{A}$ and node feature matrix $\mathbf{F}$, we adopt the state-of-the-art method GCN proposed in [Kipf and Welling, 2016] to learn the discriminative network representations $\mathbf{R}$. Any other semi-supervised network representation learning methods can be readily incorporated into ANRMAB.

The layer-wise propagation rule of GCN is defined as follows

$$\mathbf{H}^{(l+1)} = \sigma(\widetilde{\mathbf{D}}^{-\frac{1}{2}} \widetilde{\mathbf{A}} \widetilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(l)} \mathbf{W}^{(l)}) \tag{7}$$

where $\widetilde{\mathbf{A}}$ is set as $\widetilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}_N$ with $\mathbf{I}_N$ being the identify matrix, $\widetilde{\mathbf{D}}_{ii} = \sum_j \widetilde{\mathbf{A}}_{ij}$ and $\mathbf{W}^{(l)}$ is a layer-specific trainable weight matrix. We use ReLU$(\cdot)$ as the activation function $\sigma(\cdot)$. $\mathbf{H}^{(l)}$ is the matrix of activations in l-th layer. $\mathbf{H}^{(0)}$ is initialed as $\mathbf{F}$. We can calculate $\widehat{\mathbf{A}} = \widetilde{\mathbf{D}}^{-\frac{1}{2}} \widetilde{\mathbf{A}} \widetilde{\mathbf{D}}^{-\frac{1}{2}}$ in a preprocessing step.

The activation function in the last layer is the softmax function, defined as softmax$(x_i) = \frac{1}{\Lambda}\exp(x_i)$ with $\Lambda = \sum_i \exp(x_i)$, which is applied row-wise. The loss function is defined as the cross-entropy error over all labeled nodes, $loss = -\sum_l \sum_{c=1}^{C} \mathbf{Y}_{lc} \ln \mathbf{Z}_{lc}$, where $\mathbf{Y}_{lc}$ is the indicator of node v_l belongs to label c and $\mathbf{Z}$ is the output of GCN. Take a two-layer model for node classification as an example, the model has the following formulation:

$$\mathbf{Z} = f(\mathbf{A}, \mathbf{F}) = \text{softmax}(\widehat{\mathbf{A}}\ \text{ReLU}(\widehat{\mathbf{A}}\mathbf{F}\mathbf{W}^{(0)})\ \mathbf{W}^{(1)}) \tag{8}$$

For the trained model, the matrix of activations in the l-th layer $\mathbf{H}^{(l)} (l = 1, 2, \dots)$, can be regarded as the corresponding network representations in the node embedding space. For specified l-th layer, $\mathbf{R} = \mathbf{H}^{(l)}$.

4.4 Method Summary

The proposed method ANRMAB, illustrated in Figure 1, mainly contains three parts, $i.e.$, AL query strategy component, active node selection component and active discriminative network representation component, which are detailed in Sections 4.1-4.3. In iteration t ($t = 1, \ldots, B$), the active discriminative network representation component takes network $\mathcal{G}$ with its adjacency matrix $\mathbf{A}$ and node feature matrix $\mathbf{F}$ as well as the updated nodes set $\mathcal{L}^t$ as input, and outputs the network representations $\mathbf{R}^t$ and the measures of node classification, based on which three AL query strategies, $i.e.$, information entropy ϕ^t_{IE}, node centrality ϕ^t_{NC} and information density ϕ^t_{ID} are generated. By using the reward scheme $r^t(v_i; f^t, \tau)$, ANRMAB takes a multi-armed bandit mechanism to adaptively query the most informative node v^* from the unlabeled nodes set $\mathcal{U}^t$. v_* is then added to $\mathcal{L}^t$ to form the updated training data $\mathcal{L}^{t+1}$, which is used for learning $\mathbf{R}^{t+1}$. The above process continues until the labeling budget B is reached. The pseudo-code of ANRMAB is listed in Algorithm 1.

5 Experiments

In this section, we evaluate experimental performance of the proposed method ANRMAB using node classification task on three public data sets. All the experiments are conducted on a Linux system with Intel(R) Core(TM) i7-2600 CPU @3.40GHz*8 and 10G memory.

5.1 Data Sets

We consider three public citation network data sets[1], Citeseer, Cora and Pubmed [Sen *et al.*, 2008], which contain sparse bag-of-words feature vector for each document and a list of citation links between documents. Each document has a class label. We treat documents as nodes and the citation links as the edges. Table 2 summarizes the statistics of the data sets.

Data set	Nodes	Edges	Classes	Features	Label rate
Citeseer	3,327	4,732	6	3,703	0.036
Cora	2,708	5,429	7	1,433	0.052
Pubmed	19,717	44,338	3	500	0.003

Table 2: Statistics of data sets.

5.2 Baseline Methods and Evaluation Metrics

We compare **ANRMAB** with the following baselines and its variants: 1) **GCN** [Kipf and Welling, 2016] is the state-of-the-art semi-supervised network representation algorithm, which is used to randomly select the node to query in AL setting; 2) **AGE** [Cai *et al.*, 2017] is the active graph embedding method which combines the AL query strategies linearly using the time-sensitive parameters; 3) **ANRMAB-entropy** is the variant of our method that excludes the information entropy query strategy; 4) **ANRMAB-centrality** is the variant that excludes the node centrality strategy; 5) **ANRMAB-density** is the variant that excludes the information density strategy.

Two popular metrics, Macro-F1 and Micro-F1 [Perozzi *et al.*, 2014], are adopted for performance evaluations.

5.3 Experimental Settings

For fair comparison, we follow the same experimental settings as in [Kipf and Welling, 2016; Cai *et al.*, 2017]. For each data set, we use 1,000 labeled nodes as the testing set for evaluation. To ensure that the performance variation in the experiments is due to different AL query strategies and their selections, we randomly sample 500 labeled nodes from the non-testing nodes for validation, which is repeated for 10 times. Label rate in Table 2 represents a maximum amount of labeled nodes (denoted as L_{max}) that are used for training divided by the total number of nodes in each data set. The discriminative network representation learning in all the compared methods are trained using Eq. (8) for a maximum of 600 epochs using Adam [Kingma and Ba, 2014] with a learning rate of 0.01, a hidden layer size of 32 and early stopping with a window size of 10.

In the AL scenario, we set the labeling budget as $B = L_{max} - L_{init}$ for each data set, where L_{init} is the number of initial labeled nodes. Considering the label balance across classes, we randomly initialize 4 labeled nodes for each class. We repeat the process for 10 times and report the average results for all experiments.

[1]https://linqs.soe.ucsc.edu/data

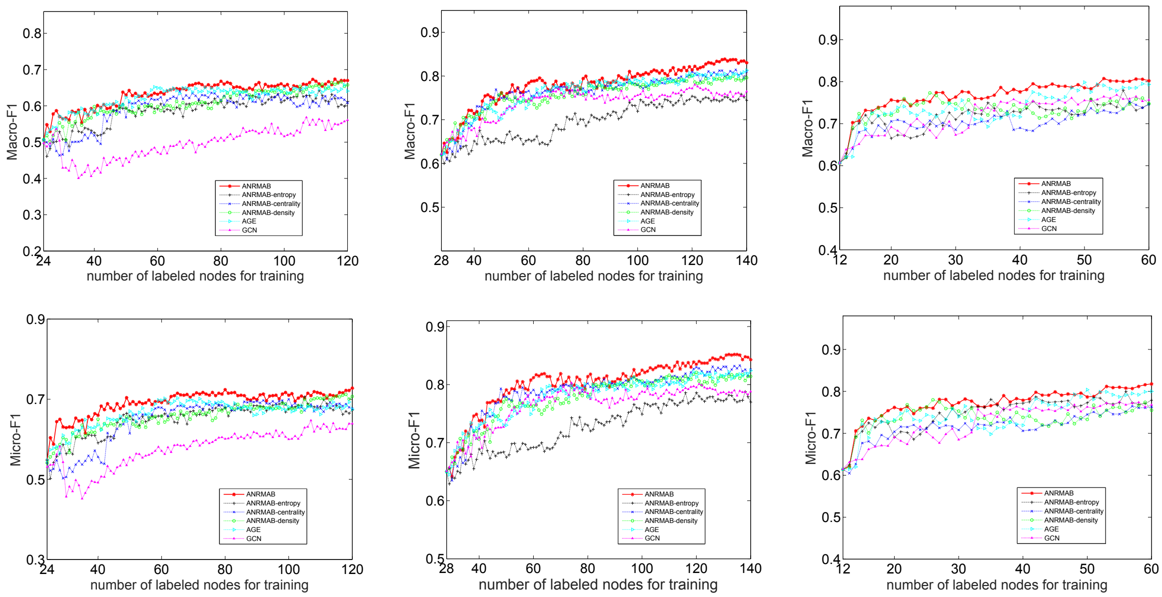
Method	Metric	Citeseer	Cora	Pubmed
ANRMAB-entropy	Macro-F1	58.6792	69.7020	71.7181
	Micro-F1	65.0701	72.8204	74.7122
ANRMAB-centrality	Macro-F1	59.1749	76.2886	70.2872
	Micro-F1	64.7959	78.7062	71.6714
ANRMAB-density	Macro-F1	60.6507	75.7604	73.1998
	Micro-F1	65.7485	77.7336	73.7939
AGE	Macro-F1	62.2396	76.1466	73.8930
	Micro-F1	66.4701	78.0230	74.4755
GCN	Macro-F1	49.5945	73.8703	71.9348
	Micro-F1	57.6680	76.5398	72.4490
ANRMAB	Macro-F1	**63.2939**	**77.9796**	**76.6087**
	Micro-F1	**69.2052**	**80.2168**	**77.1082**

Table 3: The Macro-F1 (%) and Micro-F1 (%) performance averaged over different number of labeled nodes for training.

5.4 Results

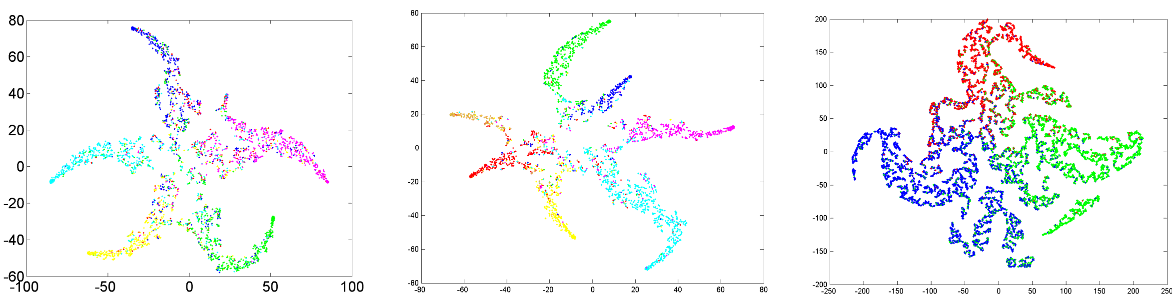
Figure 2 shows the Macro-F1 and Micro-F1 performance comparisons with different number of labeled nodes for training, for which each compared method queries a corresponding informative node added to the training data in each iteration. From the figures, we see that with the increasing number of labeled nodes for training, the Macro-F1 and Micro-F1 values show an overall upward trend as more label information is considered. No variant method of ANRMAB outperforms other variants across the three data sets and the span of sizes of training labeled nodes sets, which shows that no single or specific combinational module can satisfy the needs of all data sets and the informative nodes should be queried adaptively.

Considering the above observations, our method



(a) Citeseer (b) Cora (c) Pubmed

Figure 2: The Macro-F1 and Micro-F1 performance comparisons using different number of labeled nodes for training (varying from the number of initial labeled nodes L_{init} to the maximum amount L_{max}, with 1 as the increment). Our method ANRMAB is marked as red line.



(a) Citeseer (b) Cora (c) Pubmed

Figure 3: t-SNE visualization of network representations using the matrix of activations in the last layer trained by ANRMAB on L_{max} labeled nodes.

ANRMAB significantly outperforms its peers in most cases across all the data sets. Table 3 shows the Macro-F1 and Micro-F1 values averaged over all the cases, from which we see that ANRMAB obtains the best performance. For example, compared with AGE and GCN, ANRMAB improves the value of Macro-F1 by 1.87% and 7.49%, Micro-F1 by 2.53% and 6.62% that averaged over all the data sets. The above experimental results demonstrate the effectiveness of ANRMAB.

To illustrate the discrimination of the learned network representations, by tuning the parameters setting of t-SNE [Maaten and Hinton, 2008], Figure 3 gives an interesting visualization using the matrix of activations in the last layer trained by ANRMAB on L_{max} labeled nodes. Though the nodes marked as red in Figure 3(a) are scattered, which account for 7.48% of nodes in Citeseer data set, the visualization results of ANRMAB for all the data sets are quite clear, with meaningful layout for each class.

6 Conclusions

In this paper, we propose a novel method ANRMAB to learn the discriminative network representations under the active learning setting to improve the performance, considering that different training labeled nodes lead to different results. We argue that no single or specific combinational module can satisfy the needs of all data distributions. By using three AL query strategies, ANRMAB incorporates a multi-armed bandit mechanism for adaptive decision making with an effective reward scheme. We conduct extensive experiments on three public data sets to demonstrate the effectiveness of ANRMAB.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments and suggestions. This work was supported by the National Key Research and Development Program of China (No.2016YFB0801301), the NSFC (No.61502479), the MQNS Grant (No.9201701203), the MQ Enterprise Partnership Scheme Pilot Res Grant (No. 9201701455), and the Youth Innovation Promotion Association CAS (No. 2017210). J. Wu is the corresponding author.

References

[Aggarwal *et al.*, 2014] Charu C Aggarwal, Xiangnan Kong, Quanquan Gu, Jiawei Han, and Philip S Yu. Active learning: A survey. 2014.

[Auer *et al.*, 2002] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002.

[Beygelzimer *et al.*, 2009] Alina Beygelzimer, Sanjoy Dasgupta, and John Langford. Importance weighted active learning. In *ICML*, pages 49–56, 2009.

[Beygelzimer *et al.*, 2011] Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandit algorithms with supervised learning guarantees. In *AISTATS*, pages 19–26, 2011.

[Cai *et al.*, 2017] HongYun Cai, Vincent Wenchen Zheng, and Kevin Chen-Chuan Chang. Active learning for graph embedding. *CoRR*, abs/1705.05085, 2017.

[Cai *et al.*, 2018] Hongyun Cai, Vincent W Zheng, and Kevin Chang. A comprehensive survey of graph embedding: problems, techniques and applications. *IEEE Transactions on Knowledge and Data Engineering*, 2018.

[Cavallari *et al.*, 2017] Sandro Cavallari, Vincent W Zheng, Hongyun Cai, Kevin Chen-Chuan Chang, and Erik Cambria. Learning community embedding with community detection and node embedding on graphs. In *CIKM*, pages 377–386, 2017.

[Donmez *et al.*, 2007] Pinar Donmez, Jaime Carbonell, and Paul Bennett. Dual strategy active learning. In *ECML*, pages 116–127, 2007.

[Ganti and Gray, 2012] Ravi Ganti and Alexander G. Gray. UPAL: unbiased pool based active learning. In *AISTATS*, pages 422–431, 2012.

[Gao *et al.*, 2016] Li Gao, Jia Wu, Zhi Qiao, Chuan Zhou, Hong Yang, and Yue Hu. Collaborative social group influence for event recommendation. In *CIKM*, pages 1941–1944, 2016.

[Gao *et al.*, 2017] Li Gao, Jia Wu, Chuan Zhou, and Yue Hu. Collaborative dynamic sparse topic regression with user profile evolution for item recommendation. In *AAAI*, pages 1316–1322, 2017.

[Gittins *et al.*, 1989] J. Gittins, K. Glazebrook, R. Weber, J. Gittins, and R. Weber. Multi-armed bandit allocation indices. *Journal of the Operational Research Society*, 40(12):1158–1159, 1989.

[Hsu and Lin, 2015] Wei-Ning Hsu and Hsuan-Tien Lin. Active learning by learning. In *AAAI*, pages 2659–2665, 2015.

[Kingma and Ba, 2014] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[Konyushkova *et al.*, 2017] Ksenia Konyushkova, Raphael Sznitman, and Pascal Fua. Learning active learning from data. In *NIPS*, pages 4228–4238, 2017.

[Maaten and Hinton, 2008] Laurens Van Der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[Newman, 2010] Mark Newman. *Networks: an introduction*. Oxford university press, 2010.

[Page *et al.*, 1999] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[Perozzi *et al.*, 2014] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: online learning of social representations. In *KDD*, pages 701–710, 2014.

[Sen *et al.*, 2008] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Gallagher, and Tina Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93–106, 2008.

[Settles and Craven, 2008] Burr Settles and Mark Craven. An analysis of active learning strategies for sequence labeling tasks. In *EMNLP*, pages 1070–1079, 2008.

[Settles, 2010] Burr Settles. Active learning literature survey. *University of Wisconsin, Madison*, 52(55-66):11, 2010.

[Stephenson and Zelen, 1989] Karen Stephenson and Marvin Zelen. Rethinking centrality: Methods and examples. *Social networks*, 11(1):1–37, 1989.

[Tu *et al.*, 2016] Cunchao Tu, Weicheng Zhang, Zhiyuan Liu, and Maosong Sun. Max-margin deepwalk: Discriminative learning of network representation. In *IJCAI*, pages 3889–3895, 2016.

[Vermorel and Mohri, 2005] Joannès Vermorel and Mehryar Mohri. Multi-armed bandit algorithms and empirical evaluation. In *ECML*, pages 437–448, 2005.

[Yang *et al.*, 2016] Zhilin Yang, William Cohen, and Ruslan Salakhudinov. Revisiting semi-supervised learning with graph embeddings. In *ICML*, pages 40–48, 2016.

[Zhang *et al.*, 2017] Ye Zhang, Matthew Lease, and Byron C. Wallace. Active discriminative text representation learning. In *AAAI*, pages 3386–3392, 2017.

[Zhu *et al.*, 2003] Xiaojin Zhu, John Lafferty, and Zoubin Ghahramani. Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In *ICML-workshop*, volume 3, 2003.