Long-short Distance Aggregation Networks for Positive Unlabeled Graph Learning

Man Wu
Florida Atlantic University
mwu2019@fau.edu

Shirui Pan
Monash University
shirui.pan@monash.edu

Lan Du
Monash University
lan.du@monash.edu

Ivor Tsang
University of Technology Sydney
ivor.tsang@uts.edu.au

Xingquan Zhu
Florida Atlantic University
xzhu3@fau.edu

Bo Du
Wuhan University
remoteking@whu.edu.cn

ABSTRACT

Graph neural nets are emerging tools to represent network nodes for classification. However, existing approaches typically suffer from two limitations: (1) they only aggregate information from short distance (e.g., 1-hop neighbors) each round and fail to capture *long distance relationship* in graphs; (2) they require users to label data from several classes to facilitate the learning of discriminative models; whereas in reality, users may only provide labels of a small number of nodes in a single class. To overcome these limitations, this paper presents a novel long-short distance aggregation networks (LSDAN) for positive unlabeled (PU) graph learning. Our theme is to generate multiple graphs at different distances based on the adjacency matrix, and further develop a long-short distance attention model for these graphs. The short-distance attention mechanism is used to capture the importance of neighbor nodes to a target node. The long-distance attention mechanism is used to capture the propagation of information within a localized area of each node and help model weights of different graphs for node representation learning. A non-negative risk estimator is further employed, to aggregate long- short-distance networks, for PU learning using back-propagated loss modeling. Experiments on real-world datasets validate the effectiveness of our approach.

CCS CONCEPTS

• **Computing methodologies → Machine learning**.

KEYWORDS

Positive unlabeled learning, Graph neural networks

ACM Reference Format:
Man Wu, Shirui Pan, Lan Du, Ivor Tsang, Xingquan Zhu, Bo Du. 2019. Long-short Distance Aggregation Networks for Positive Unlabeled Graph Learning. In *the 28th ACM International Conference on Information and Knowledge Management (CIKM'19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3357384.3358122

CIKM '19, November 3–7, 2019, Beijing, China

ACM ISBN 978-1-4503-6976-3/19/11…$15.00
https://doi.org/10.1145/3357384.3358122

1 INTRODUCTION

Graphs are fundamental tools to model inter-dependence among data in many applications including social media networks, citation networks, and protein-protein interaction networks. However, graph data is naturally sparse and highly complicated, which makes the node classification task profoundly difficult.

To enable node classification in graphs, recent approaches have proposed to focus on learning a new representation which embeds both structure and node content information in a compact and low dimensional space. The graph neural network approaches, graph convolutional networks (GCNs) [2] in particular, have achieved impressive performance in recent years. The basic idea of GCNs is to develop a *convolutional* layer which aggregates the attributes from neighbor nodes to a target node iteratively to guide the classification task in an attributed graph. In GCNs [8], the *aggregation* is defined as the average or summarization of neighboring feature information, which considers the importance of each neighbor equally in the learning process. Recently, the graph attention network (GAT) is proposed to learn the weights of different neighbors for information aggregation [7]. However, one major limitation of GAT is that they only exploit the direct (1-hop) neighbor nodes for attention learning. *Long distance relationship* is largely ignored in each iteration. In practice, *long distance relationship* is vitally important. For instance, in a real social network, each individual is a member of several communities and can be influenced by her/his neighborhoods with different distances around her/him, ranging from short distance relationship (e.g. families, friends), to long distance relationship (e.g. society, nation states). Every single relationship is usually sparse and biased, thus long distance relationship should be also considered to obtain a comprehensive representation of the node for graph learning collaboratively.

Another drawback of existing graph neural nets is that they require users to label data from several classes to facilitate the classification task. In reality, users may only provide the labels of interest in a single class for a small number of nodes. Taking Internet surfing as an example, the Internet is a huge graph, in which users many only bookmark pages they are interested in (i.e. the positive data), ignoring a large amount of other pages (i.e. the unlabeled data). Accurately recommending pages or news interesting to users, according to their bookmarks, is a positive unlabeled learning problem. In this paper, we study the problem of positive unlabeled graph learning, where only a small portion of positive nodes are labeled. Considering the GCNs as the learning

framework, as popularly used in previous works [2], we summarize the challenges as follows,

- *Challenge 1*: How can we capture graph structure information with long-distance neighbors? Existing graph neural networks typical only utilize short-distance information in a single layer.
- *Challenge 2*: How to design an end to end framework for positive unlabeled graph learning? Current GCNs require class labels from several classes to learn a model.

To overcome the above challenges, we propose a novel long-short distance aggregation network (LSDAN) for positive unlabeled (PU) learning for graphs. For *Challenge 1*, we first generate multiple graphs in different hops based on the adjacency matrix, then develop a long-short distance attention model for these graphs. The long-short distance attention model employs a short-distance attention mechanism to capture the importance of each neighbor node to a target node, and utilizes a long-distance attention approach to model the weights of the different graph with different neighbor nodes for the representation learning. For *Challenge 2*, we employ a *non-negative risk estimator* for PU learning and the expected loss is back-propagated for model learning. Experimental results on real datasets validate the design and effectiveness of our approach. Our contributions can be summarized below:

- We first study the problem of *positive unlabeled graph learning* for network node classification, and present a new deep learning model LSDAN as a solution.
- We propose a novel attention network for graph data, which captures node significance in both short-distance and long-distance graphs, to model the long-short distance neighboring information in a single layer.
- Experiments on benchmark graph datasets demonstrate that our approach outperforms baseline methods.

2 PROBLEM STATEMENT

Graph: A graph is represented as $G = (V, E, X, Y)$, where $V = \{v_i\}_{i=1,\cdots,N}$ is a vertex set representing the nodes in a graph, and $e_{i,j} = (v_i, v_j) \in E$ is an edge indicating the relationship between two nodes. The topological structure of graph G can be represented by an adjacency matrix A, where $A_{i,j} = 1$ if $(v_i, v_j) \in E$; otherwise $A_{i,j} = 0$. $x_i \in X$ indicates content features associated with each node v_i. $y_i \in Y = \{+1, -1\}$ is the ground-truth class label for each node, where if a node v_i is of interest of a user, then $y_i = 1$ otherwise $y_i = -1$.

Positive Unlabeled Graph Learning (PUGL): Assume $V = P \bigcup U$, where P are the labeled nodes ($\forall v_i \in P$, $y_i = 1$) and U are unlabeled nodes. Given a graph $G = (V, E, X, Y)$, Positive Unlabeled Graph Learning (PUGL) **aims** to learn a binary classifier model, $f : (A, X; P) \mapsto Y$, to predict the class labels for the unlabeled nodes U. In this paper, we propose the first deep learning model for PUGL.

3 LONG-SHORT DISTANCE AGGREGATION NETWORKS FOR PU GRAPH LEARNING

In this section, we will present our proposed LSDAN algorithm for PU Graph learning. Our learning objectives are to (1) capture the *long-short distance relationship* between nodes, and (2) enable PU learning in a graph. We will first present our long-short distance attention network which exploits both short-distance and long-distance attention for *long-short distance relationship* modeling. Then we present our unbiased risk estimator for PU learning. Our framework, as shown in Figure 1, mainly consists of three components.

- **Short-Distance Attention**. For the input X and an adjacent matrix A, a short-distance self attention mechanism is applied to learn a representation for each node.
- **Long-short Distance Attention**. Given an input graph G, we will first generate multi-hop graph representation based on adjacent matrix $A^1, A^2, \cdots, A^K$. The matrix A^k captures the neighbors in the k-th hop of the graph G. We develop a long-distance attention approach to automatically determine the weights of different graphs $A^1, A^2, \cdots, A^K$.
- **Unbiased PU Learning**. Based on our long-short distance attention model, we develop a deep architecture for learning the graph representation of each node. Then a non-negative risk estimator is used to estimate the classification loss. The loss is further back-propagated to the learning progress in an end to end learning framework.

3.1 Short-Distance vs. Long-Distance

Definition 1. (Short-Distance). *Short-distance is defined as the distance from direct (1-hop) neighbor nodes to a target node.*

The (normalized) adjacency matrix A characterizes the first-order proximity to model the direct relationship between vertices.

Definition 2. (Long-Distance). *Long-distance is defined as the distances from k-hop neighbors ($k > 1$) to a target node.*

Given an input graph G, we will first generate multi-hop graph representation based on adjacent matrix $A^1, A^2, \cdots, A^K$. The matrix $A^k = \underbrace{A \cdot A \cdots A}_{k}$ is the matrix power of A, i.e., matrix product of k copies of A, which captures the neighbors in the k-th hop of the graph G. Specifically, $A^k_{i,j} > 0$ indicates there are some path from v_i to v_j through extract k-hop.

3.2 Long-short Distance Attention

Short-Distance Attention Given the input X and an adjacent matrix A, a short-distance self attention mechanism is applied to learn a representation for each node, which can better capture the node features of the whole graph with short distance. Specifically, we will have

$$h_i = g\Big(\sum_{j \in \Gamma_i} \alpha_{i,j} W x_j\Big), \tag{1}$$

where g is a non-linear activation function, Γ_i is the short-distance neighbors for node v_i, and $\alpha_{i,j}$ is weight value for each neighbor v_j. To compute $\alpha_{i,j}$, a shared linear transformation is applied to each node through multiply a shared weight matrix $W \in \mathbb{R}^{D \times M}$ in the initial step. Then $\alpha_{i,j}$ is computed by an attention function $\texttt{Att}$: $\mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$

$$\alpha_{i,j} = softmax\big(\texttt{Att}(Wx_i, Wx_j)\big), \tag{2}$$

which measures the importance of vertex j to vertex i. Here the attention mechanism $\texttt{Att}$ is instantiated with a dot product (parametrized by a weight vector $r \in \mathbb{R}^{2D}$) and a LeakyReLU nonlinearity.

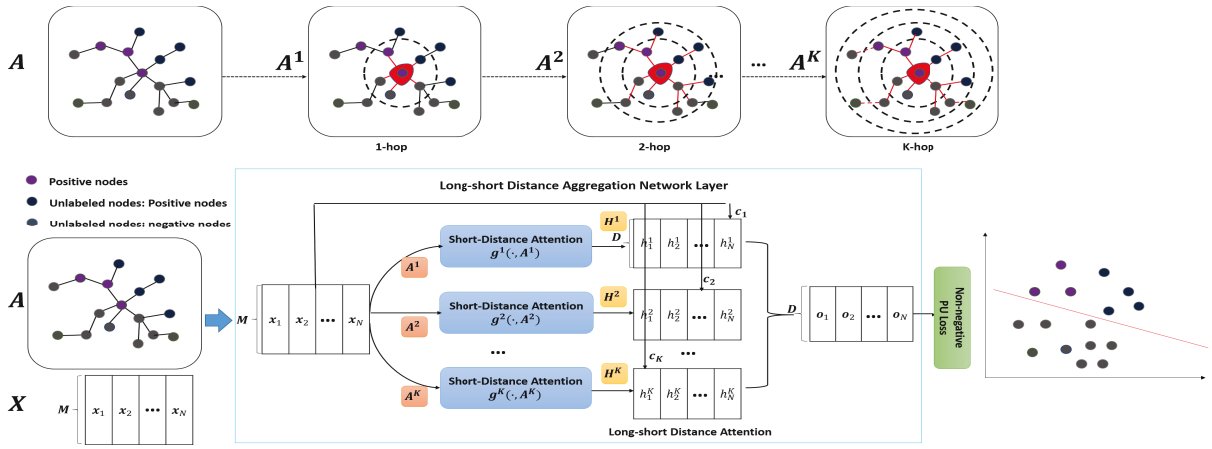


Figure 1: The overall architecture of the proposed long-short distance aggregation network (LSDAN) model. Upper panel: LSDAN uses higher order adjacency matrices to capture *long distance relationship w.r.t.* a target node. Lower panel: LSDAN uses higher-order network topology structures and node content (X) to progressively learn a long-short distance attention model, whose outputs are integrated into a learning objective function to achieve optimized PU graph learning outcomes.

Long-short Distance Attention We aggregate embedding from different graphs to produce a unified representation. As neighbors from different distances contribute differently to learning the representation, we propose an *Long-Distance Attention* scheme to capture the significance of each graph.

Specially, for each $A^k, k \in \{1, \cdots, K\}$, we will perform the short-distance self attention to learn the embedding H^k for each node. We then use the original input X as the key of the attention mechanism, and perform attention on each graph output H^k, an attention coefficient c_i^k is computed by an attention function f:

$$c_i^k = f(h_i^k, Jx_i^k), \tag{3}$$

where J is a shared weight matrix to make the input x_i^k of node i have the same dimension with the output h_i^k. Then we further normalize the weight c_i^k with a softmax layer.

$$c_i^k = \frac{exp\left(c_i^k\right)}{\sum_{k=1}^{K} exp\left(c_i^k\right)}, \tag{4}$$

After implementing the attention, we can get the final output $O = \{o_1, \cdots, o_N\}, o_i \in \mathbb{R}^D$:

$$o_i = \sum_{k=1}^{K} c_i^k h_i^k. \tag{5}$$

The short-distance attention and long-distance attention components are integrated into a unified layer, *Long-short Distance Aggregation Network Layer* (LSDAN), which serves as a building block to construct a deep graph neural network.

3.3 Positive and Unlabeled Graph Learning

Using above long-short distance aggregation network, we obtain the new representation $O^L = \{o_1^L, \cdots, o_N^L\}, o_i^L \in \mathbb{R}^2$ in the final layer. To facilitate PU learning, we minimize a *non-negative risk*.

We denote $\pi_p = p(Y = +1)$ be the *class-prior probability*, $\pi_n = p(Y = -1) = 1 - \pi_p$. π_p is assumed to known throughout the paper; it can be estimated from positive (P) and Unlabel (U) data [1]. Let $\mathcal{L} : \mathbb{R} \times \{\pm 1\} \to \mathbb{R}$ be a loss function, then $\mathcal{L}(y', y)$ measures the predicting loss for an output y' when the ground truth is y. Let $s(o)$ be a sigmoid function to map the input o in the range (0,1) . Motivated by Kiryo et al. [3], we employ a non-negative risk estimator $\hat{R}_{pu}(s)$, given as follows,

$$\tilde{R}_{pu}(s) = \pi_p \hat{R}_p^+(s) + \max\left\{0, \hat{R}_u^-(s) - \pi_p \hat{R}_p^-(s)\right\}. \tag{6}$$

where $\hat{R}_p^+(s) = (1/n_p) \sum_{i=1}^{n_p} \mathcal{L}(s(o_i^p), +1)$ and $\hat{R}_p^-(s) = (1/n_p) \sum_{i=1}^{n_p} \mathcal{L}(s(o_i^p), -1)$ are the approximated risks for positive samples, and $\hat{R}_u^-(s) = (1/n_u) \sum_{i=1}^{n_u} \mathcal{L}(s(o_i^u), -1)$ is the risk for negative samples.

By minimising the risk via Eq.(6), our model can be learned in an end to end manner. The expected loss/risk is back-propagated to guide the representation learning for better PU graph learning.

4 EXPERIMENTS

Datasets We employ two widely used citation network datasets (Citeseer, DBLP) for node classification [5, 9]. As these datasets have multiple classes, we select one class as P (positive) class, and all the other classes are regarded as N (negative) class, through which we convert the classification problem on each dataset into a binary classification problem.

Baselines To the best of our knowledge, there is no study on positive unlabeled graph learning. To make a fair comparison and evaluate the effectiveness of our design, we select the following baselines with necessary adaption.

- **OC-SVM** [6] is the One-class SVM algorithm which uses only positive examples from the node content for learning.
- **Roc-SVM** [4] uses two step strategies to build a classifier from the node content.
- **FS-PU:** Full-connected self-attention network PU (FS-PU) uses the node features with a self-attention network.
- **F-PU:** Full-connected PU (FS-PU) only uses the node features with a multiple layer perceptron (MLP).

- **GCN** uses the graph convolutional network [2] to integrate structure and content.
- **GAT** uses the graph attention nets [7] to exploit structure and content.

Note that we have integrated the *non-negative risk estimator* into GCN-PU, GAT-PU, FS-PU, F-PU to faciliate PU learning.

Experimental Setup For fairness of comparison, we randomly split each PN dataset into the positive and unlabeled set. Following Kiryo et al. [3], we sample N_{PN} (the total number of positive nodes) nodes from N as negative class. Then we select $p * N_{PN}$ nodes from P as the training set, the rest positive nodes and negative nodes are used as the unlabeled set (p is the percentage of training (positive) nodes). We conduct 10 trials of randomly splitting, and report the average **F1 score** as final experimental results. All models were implemented in TensorFlow. For the proposed LSDAN, the number hops K is set to 4.

Experimental Results The results of our evaluation experiments are presented in Table 1, Table 2. From these results, we have the following observations:

(1) The OC-SVM and Roc-SVM obtain worse performance than other methods. This is because the traditional shallow methods do not capture the graph structure information. The GAT obtains better performance than F-PU and FS-PU, which shows that it is useful to learn the node representation by introducing the relationships of nodes in PU learning.
(3) The proposed LSDAN outperforms GAT which only captures short-distance information. The results show the effectiveness of our algorithm in exploiting multi-hop neighbors to capture *long-short distance relationship* in graph learning.
(4) The results also show that LSDAN consistently outperforms all the other baselines on all three datasets with different training ratios.It demonstrates that long-short distance aggregation network together with the non-negative risk estimator can better capture data distribution and the underlying relationship among data by integrating the feature information and graph information into a unified framework.

Parameter Analysis

Embedding Dimensions D: We vary D with $\%p = 2$, $L = 2$ and report the results on the two datasets in Fig. 2(a). We can find that the F1 score shows an apparent increase from 8 to 64 in the DBLP, while it decreases slightly in the 32nd dimension in the Citeseer. When the number of embedding dimensions continuously increases, the performance starts to remain stable. This is intuitive as more dimensions can encode more useful information from data.

Distance at K-Hops : We also report the F1 scores over different choices of K with $\%p = 2$ and $L = 2$ in Fig. 2(b). We can observe that the setting $K = 2$ has a significant improvement over the setting $K = 1$ on two datasets. This confirms that the long distance relation is really important to better capture graph structure information, and multiple graphs can learn complementary local information. When K is large enough, we can find that learned K-hop relational information becomes weak and shifts towards a steady result.

5 CONCLUSIONS

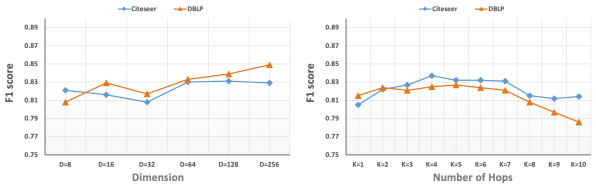
In this paper, we propose a novel long-short distance aggregation network (LSDAN) for positive unlabeled graph learning. We argued that existing algorithms only exploit 1-hop neighbors to aggregate information to learn the representation for nodes, which largely overlook the *long-distance relationship*. To this end, we proposed a long-short distance aggregation network to jointly exploit the short-distance and long-short attention from multiple graphs (representation of graph). We further employed a novel non-negative risk estimator for positive unlabeled graph learning in an end to end framework. The results on real-world graph datasets demonstrate the effectiveness of our algorithm.

Table 1: The F1 score on Citeseer.

%p	OC-SVM	Roc-SVM	F-PU	FS-PU	GCN	GAT	LSDAN
1	0.023	0.018	0.684	0.682	0.433	0.775	**0.786**
2	0.038	0.057	0.626	0.695	0.564	0.775	**0.804**
3	0.054	0.079	0.710	0.705	0.623	0.796	**0.813**
4	0.090	0.115	0.734	0.725	0.721	0.814	**0.828**

Table 2: The F1 score on DBLP.

%p	OC-SVM	Roc-SVM	F-PU	FS-PU	GCN	GAT	LSDAN
1	0.445	0.056	0.650	0.677	0.419	0.767	**0.808**
2	0.543	0.144	0.521	0.695	0.599	0.807	**0.833**
3	0.580	0.234	0.710	0.715	0.685	0.824	**0.824**
4	0.601	0.314	0.597	0.725	0.734	0.836	**0.849**



(a) Dimension (b) Distances at k-hops

Figure 2: Parameter analysis on D and K-hops

ACKNOWLEDGMENTS

This research is supported by the US National Science Foundation (NSF) through Grants IIS-1763452 and CNS-1828181, and Australian Research Council through grants LP150100671 and DP180100106. S. Pan is the corresponding author.

REFERENCES

[1] Shantanu Jain, Martha White, and Predrag Radivojac. 2016. Estimating the class prior and posterior from noisy positives and unlabeled data. In *NIPS*. 2693–2701.
[2] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).
[3] Ryuichi Kiryo, Gang Niu, Marthinus C du Plessis, and Masashi Sugiyama. 2017. Positive-unlabeled learning with non-negative risk estimator. In *NIPS*. 1675–1685.
[4] Xiaoli Li and Bing Liu. 2003. Learning to classify texts using positive and unlabeled data. In *IJCAI*. 587–592.
[5] Shirui Pan, Jia Wu, Xingquan Zhu, Chengqi Zhang, and Yang Wang. 2016. Tri-party deep network representation. In *IJCAI*. 1895–1901.
[6] Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. 2014. Estimating the Support of a High-Dimensional Distribution. *Neural Computation* 13, 7 (2014), 1443–1471.
[7] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).
[8] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S Yu. 2019. A comprehensive survey on graph neural networks. *arXiv:1901.00596* (2019).
[9] Cheng Yang, Zhiyuan Liu, Deli Zhao, Maosong Sun, and Edward Y Chang. 2015. Network representation learning with rich text information.. In *IJCAI*. 2111–2117.