Universal Domain Adaptive Network Embedding for Node Classification

Jushuo Chen
Institute of Information Engineering, Chinese Academy of Sciences & School of Cyber Security, University of Chinese Academy of Sciences
Beijing, China
chenjushuo@iie.ac.cn

Feifei Dai
Institute of Information Engineering, Chinese Academy of Sciences
Beijing, China
daifeifei@iie.ac.cn

Xiaoyan Gu*
Institute of Information Engineering, Chinese Academy of Sciences & School of Cyber Security, University of Chinese Academy of Sciences
Beijing, China
guxiaoyan@iie.ac.cn

Jiang Zhou
Institute of Information Engineering, Chinese Academy of Sciences
Beijing, China
zhoujiang@iie.ac.cn

Bo Li
Institute of Information Engineering, Chinese Academy of Sciences
Beijing, China
libo@iie.ac.cn

Weiping Wang
Institute of Information Engineering, Chinese Academy of Sciences
Beijing, China
wangweiping@iie.ac.cn

ABSTRACT

Cross-network node classification aims to leverage the abundant knowledge from a labeled source network to help classify the node in an unlabeled target network. However, existing methods assume that label sets are identical across domains, which is easily violated in practice. Hence, we attempt to integrate network embedding with universal domain adaptation, which transfers valuable knowledge across domains without assumption on the label sets, to assist in node classification. Nonetheless, the complex network relationships between nodes increase the difficulty of this universal domain adaptive node classification task. In this work, we propose a novel Universal Domain Adaptive Network Embedding (UDANE) framework, which learns transferable node representations across networks to succeed in such a task. Technically, we first adopt the cross-network node embedding component to model comprehensive node information of both networks. Then we employ the inter-domain adaptive alignment component to exploit and relate knowledge across domains, learning domain-invariant representation for knowledge transfer. In addition, the intra-domain contrastive alignment component is proposed to learn discriminative representations beneficial for classification by sufficiently utilizing unlabeled data in the target domain. Extensive experiments have been conducted on real-world datasets, demonstrating that the proposed UDANE model outperforms the state-of-the-art baselines by a large margin.

CCS CONCEPTS

• **Mathematics of computing** → **Graph algorithms**; • **Computing methodologies** → **Learning latent representations**.

*Corresponding author.

MM '23, October 29-November 3, 2023, Ottawa, ON, Canada

ACM ISBN 979-8-4007-0108-5/23/10.
https://doi.org/10.1145/3581783.3613811

KEYWORDS

Cross-network Node Classification, Universal Domain Adaptation, Network Embedding, Contrastive Learning

ACM Reference Format:
Jushuo Chen, Feifei Dai, Xiaoyan Gu, Jiang Zhou, Bo Li, and Weiping Wang. 2023. Universal Domain Adaptive Network Embedding for Node Classification. In *Proceedings of the 31st ACM International Conference on Multimedia (MM '23), October 29-November 3, 2023, Ottawa, ON, Canada.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3581783.3613811

1 INTRODUCTION

Node classification is a central task in various network scenarios, such as social networks [1], citation networks [16], and protein-protein interaction networks [5]. However, labeled data is scarce in the real world, making it hard to train an effective classifier for node classification. To address the label scarcity issue, cross-network node classification [4, 32, 33] is proposed. This task, which aims to transfer valuable knowledge from a labeled source network to help predict node labels in an unlabeled target network, has received careful attention owing to its cost-effectiveness [11].

Researchers have recently attempted to achieve cross-network node classification by learning transferable node representations. These methods are based on a strong assumption that the label sets of two networks are identical. However, practical scenarios are far more complicated, and this assumption is easily violated. For example, in citation network scenarios, given a source network from computer vision (CV) field and a target network from natural language processing (NLP) field, whose node represents an article and label indicates its research topic. As shown in Figure 1, both networks have not only common classes (e.g., "image caption," "text-image retrieval") shared by two label sets but also private classes (e.g., "object detection," "machine translation"). In this case, the assumption that label sets of two networks are identical is not valid, and existing methods would yield unsatisfactory performance.

Universal domain adaptation (UniDA) [43], which transfers knowledge across domains without assumption on the label sets, aiming not only to perform adaptation by reducing the large feature distribution discrepancy between domains but also to separate target

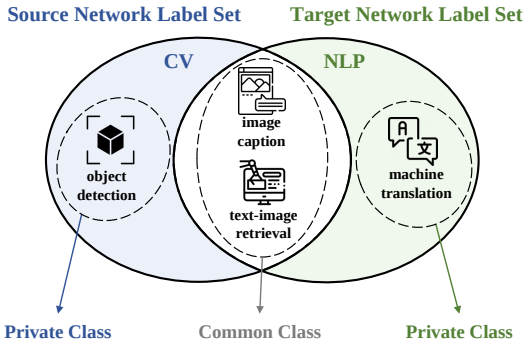


Figure 1: An example of the relationship between two network label sets. Both networks have shared common classes and unique private classes.

samples from private and common classes. Hence, UniDA is more applicable to practical scenarios. Currently, studies of UniDA mainly concentrate on CV field [6, 8, 30] and have been attempted from different aspects. For example, UAN [43] and SUDA [19] employ weighted adversarial networks to discover the common classes between domains and promote adaptation for common classes. DANCE [29] designs a neighborhood clustering objective to move each target sample to its neighbor, extracting clustered features for cross-domain classification. Even with good performance in vision domains, existing UniDA methods cannot be directly applied to network analysis. In these methods, each data sample is considered independent and identically distributed. Nonetheless, nodes in the network are connected with edges representing their relations. As such, it should be essential to consider the complex network relationships between nodes for network analysis.

In this work, we consider a universal domain adaptive node classification task. Given a source network with fully labeled nodes and a target network without labeled nodes, there are no assumptions about the relationship between the label sets of the two networks. When performing label prediction on the target network, if a node belongs to common classes, we should classify it correctly; otherwise, mark it as "unknown" (i.e., private classes of the target network are gathered as one unknown class). Since network embedding [13, 28, 36] is a powerful means to model the network structural information, we attempt to incorporate UniDA technique into network embedding for achieving universal domain adaptive node classification.

Nevertheless, developing a universal domain adaptive framework for node classification suffers from two crucial challenges: (1) *How to learn domain-invariant node representations so that valuable knowledge can be effectively transferred across domains?* In vision tasks, representation learning is only associated with each sample's absolute position in the feature space. However, network embedding requires preserving the relative position of all node pairs simultaneously. The relationships of node pairs increase the difficulty of learning domain-invariant node representations. In this circumstance, existing UniDA methods, which lack discriminability for separation and neglect the sample category during adaptation, are inadequate to exploit and relate the knowledge across domains to learn domain-invariant representations. Therefore, an effective model is called for this challenge. (2) *How to learn discriminative node representations so that target semantics can be fully explored to assist in classification?* In the UniDA setting, supervision from the source domain only ensures partial knowledge transfer. This creates aligned and unaligned sub-distributions of the target domain, which cause the intra-domain discrepancy, lowering the classification accuracy. Even worse, the complex relationships between nodes exacerbate the discrepancy. However, existing methods overly rely on source supervision and do not consider target semantic information. This makes it difficult to generate discriminative representations for the target domain to reduce the intra-domain discrepancy. As such, how to utilize the rich unlabeled target data for discriminative representation learning is a crucial challenge.

To conquer the above challenges, we propose a novel Universal Domain Adaptive Network Embedding (UDANE) framework for universal domain adaptive node classification. Our framework first models the comprehensive information of each node in cross-network scenarios. On this basis, two significant components are proposed to learn transferable node representations for classification. On the one hand, the Inter-Domain Adaptive Alignment component is proposed to learn domain-invariant node representations for knowledge transfer. Specifically, a discriminative scoring scheme with a finer separation strategy is designed to divide target data. Then this component reduces the inter-domain distribution discrepancy using prototype contrastive learning and entropy regularization. Consequently, the unlabeled target distribution adaptively aligns with the labeled source distribution to make representations domain-invariant. On the other hand, the Intra-Domain Contrastive Alignment component is proposed to learn discriminative node representations, which explores target semantics beneficial for classification. This component first generates two representations for each unlabeled node. Afterward, the alignment of representations of the same node helps to generate better target clusters. This enables discriminative representation learning for the target domain, minimizing intra-domain discrepancy in a self-supervised way. By composing the two above components, UDANE achieves better universal domain adaptive representation learning.

Our main contributions can be summarized as follows:

- We propose a novel universal domain adaptive framework UDANE for node classification. As far as we know, we are the first work to investigate universal domain adaptive node classification.
- We design a new way to learn domain-invariant and discriminative node representations using two alignment components. UDANE encourages class information transfer across networks in an adaptive and self-supervised manner.
- We evaluate our method on real-world datasets. Extensive experimental results demonstrate the superiority of UDANE over state-of-the-art baselines by a large margin, and further model analyses verify the effectiveness of UDANE.

2 RELATED WORK

2.1 Single-Network Embedding

Network embedding is aimed at mapping the vertices of a network into a low-dimensional vector space to facilitate a range of

learning tasks, such as node classification [42] and network visualization [35]. Existing methods can be categorized as transductive and inductive methods. Transductive methods usually apply matrix factorization [3, 18] or Skip-Gram model [28, 36] to optimize the representation vectors directly. Inductive methods learn functions that take the structural information and node attributes as input and output their representation vectors. They usually model the mapping function via graph convolutional networks (GCN) [16] or follow-up works such as GraphSAGE [13]. However, the single-network-based attributed network embedding algorithms without addressing domain discrepancy would have limited performance in cross-network learning scenarios.

2.2 Cross-Network Node Classification

Domain adaptation (DA) [21, 22, 25] aims to learn machine learning models transferable on different but relevant domains sharing the same label sets. Several network embedding algorithms combined with DA have been proposed for cross-network node classification. These methods can be categorized as statistic-based and adversarial-based. On the one hand, statistic-based methods [33, 40, 41, 47] incorporate the Maximum Mean Discrepancy (MMD) metric [10] into deep neural networks to match distributions across domains. On the other hand, motivated by the idea of GAN [9], adversarial-based methods [4, 44] utilize an adversarial loss to minimize the domain discrepancy. For example, ACDNE [32] utilizes two feature extractors to jointly preserve attributed and topological proximities between nodes. UDA-GCN [39] jointly exploits local and global consistency for feature aggregation and uses an attention mechanism to produce a unified representation. Nevertheless, these methods assume that the label sets of two considered networks should be identical, which is easily violated in practice. Therefore, this paper designs a novel universal domain adaptive framework for node classification without any assumption on the label sets.

2.3 Universal Domain Adaptation

Universal domain adaptation (UniDA) is a challenging DA task that removes assumptions about the relationship between source and target label sets. In recent years, UniDA has already attracted much interest from CV field [6, 30, 45]. For example, UAN [43] and SUDA [19] design metrics to divide target samples into common or private classes and employ adversarial networks to promote adaptation. DANCE [29] employs an entropy strategy to separate target samples while clustering target samples to their neighbor to consider alignment with the source. DCC [17] introduces a consensus clustering to discover discriminative clusters on common and private classes, then aligns common clusters from both domains. Nonetheless, these methods cannot be directly applied to network analysis. Thus, it should be essential to consider the complex network relationships for node classification.

3 METHODOLOGY

3.1 Problem Definition

Let $G = (V, A, X)$ denote an attributed network where V is the node set ($n = |V|$). $A \in \mathbb{R}^{n \times n}$ is the adjacency matrix to represent the network topological structure of G, where $A_{ij} = 1$ if nodes i and j are connected; otherwise $A_{ij} = 0$. $X \in \mathbb{R}^{n \times w}$ is the node attribute matrix with w as the number of node attributes in G. For the node classification task, each network G has a corresponding label matrix $Y \in \mathbb{R}^{n \times c}$ with c as the number of label categories. Moreover, the domain of G can be defined as $D = \{G, f(G)\}$, which includes G and a function $f(G)$ to predict node labels.

In the **universal domain adaptive node classification** problem, we have a fully labeled source network $G^s = (V^s, A^s, X^s)$ with associated label matrix Y^s and an unlabeled target network $G^t = (V^t, A^t, X^t)$, where $n^s = |V^s|$ and $n^t = |V^t|$. Then, we denote the source and the target network domain as D^s and D^t, respectively. Note that the source data are sampled from distribution p while the target data from distribution q. We use C_s and C_t to denote the label set of D^s and D^t, respectively. Afterward, we denote the common label set shared by both domains as $C = C_s \cap C_t$ and the private label set for two domains as $\overline{C}_s = C_s \backslash C$ and $\overline{C}_t = C_t \backslash C$, respectively. Hence, p_C is used to denote the distribution of source data with labels in C. Similarly, q_C, $q_{\overline{C}_t}$ for target distributions with labels in C, $\overline{C}_t$, respectively.

Our goal is to distinguish between target data coming from C and target data coming from $\overline{C}_t$ (i.e., "unknown" class), as well as to learn a classification model F to minimize the target risk in C, i.e., $\min \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim q_C} [F(\mathbf{x}) \neq \mathbf{y}]$. Note that G^t is fully unlabeled and C_t is inaccessible in training.

3.2 Overall Framework

To enable universal domain adaptation node classification, we propose UDANE to learn low-dimensional feature representations shared across domains. As shown in Figure 2, our framework mainly contains the following three components:

- **Cross-Domain Node Embedding.** To learn the comprehensive representation of each node, we utilize a GCN with shared weight architecture to project the nodes into a shared embedding space while preserving the cross-network structural similarity.
- **Inter-Domain Adaptive Alignment.** We first design a discriminative scoring scheme based on two signals with a finer separation strategy to divide target samples into different classes. Then we employ prototype contrastive learning to reduce the inter-domain discrepancy of common classes. Meanwhile, an entropy regularization is adopted to enforce a better separation between common and private classes.
- **Intra-Domain Contrastive Alignment.** We propose to use instance contrastive learning to utilize the rich unlabeled data in the target domain. In this, we get two representations of the unlabeled target nodes using GCN and MLP. By ensuring consistent prediction for these two representations of the same node, we minimize the intra-domain discrepancy to generate better target clusters for classification.

3.3 Cross-Network Node Embedding

Due to the complex network relationships between nodes, it should be necessary to incorporate network structures, node attributes, and node labels in a principled way to model the comprehensive information of each node from two distinct networks.

Under the structural equivalence hypothesis in complex network theory [12, 15], two nodes that have similar local network

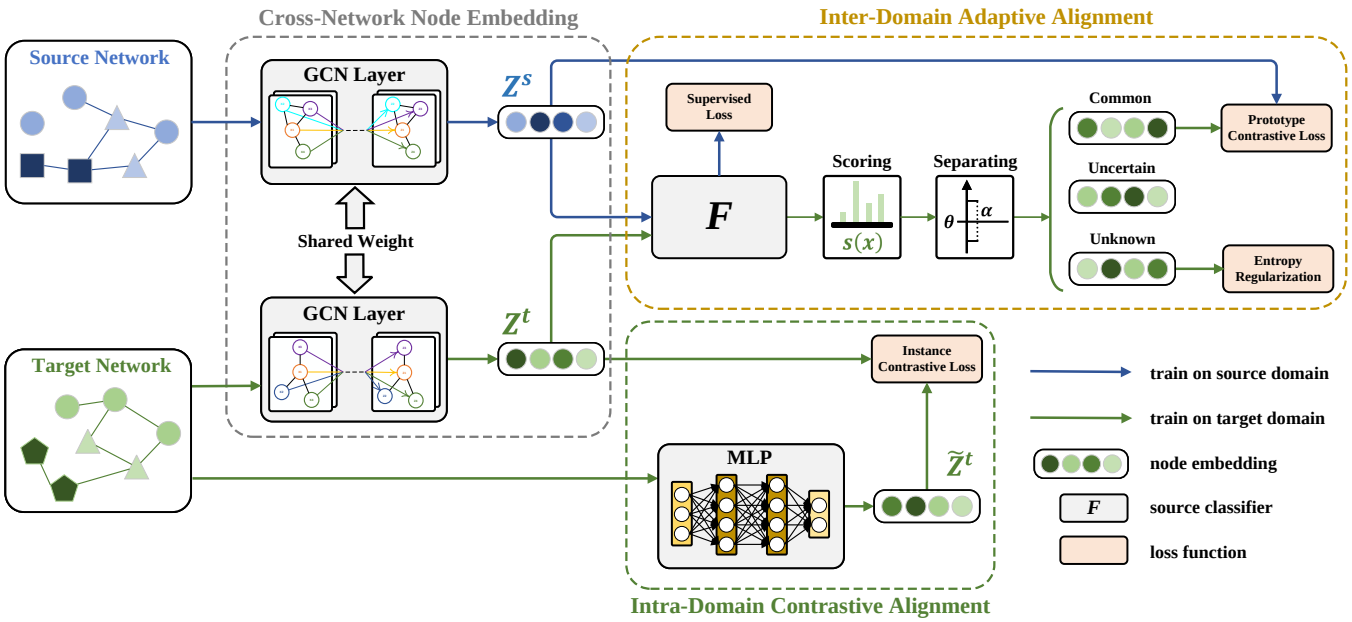


Figure 2: The overall framework of the proposed UDANE. The input consists of networks from source and target domains. Our UDANE model contains three components: (1) Cross-Network Node Embedding, (2) Inter-Domain Adaptive Alignment, and (3) Intra-Domain Contrastive Alignment.

neighborhoods can be considered structurally similar even if they are from two different networks. Therefore, we adopt a GCN with shared weight architecture to model network structures and node attributes jointly, and the proximities between nodes within and across networks can be captured. In the shared weight GCN, given input adjacency matrix A and node attribute matrix X, the output of the i-th hidden layer is constructed as:

$$Z^{(i)} = \sigma\left(\widehat{D}^{-\frac{1}{2}}\hat{A}\widehat{D}^{-\frac{1}{2}}Z^{(i-1)}W_g^{(i)}\right), \tag{1}$$

where $\hat{A} = A+I_n$ is the normalized adjacency matrix with a self-loop at each node (I_n is the identity matrix), and $\widehat{D}_{ii} = \sum_j \hat{A}_{ij}$. $Z^{(i-1)}$ is the output of the (i-1)-th layer with $Z^{(0)} = X$. σ denotes the activation function and $W_g^{(i)}$ are trainable parameters associated with the i-th hidden layer of GCN. After performing the shared weight GCN, we obtain node representations Z^s and Z^t for G^s and G^t, respectively.

3.4 Inter-Domain Adaptive Alignment

In this component, the node representations Z^s and Z^t, in turn, are fed to a node classifier F. It is noteworthy that F outputs the label prediction over C_s, thus we get $\hat{Y}^s = F\left(Z^s\right) \in \mathbb{R}^{n^s \times |C_s|}$ and $\hat{Y}^t = F\left(Z^t\right) \in \mathbb{R}^{n^t \times |C_s|}$, respectively.

To make node representations discriminative for the source domain, we use the cross-entropy error over all the labeled nodes in D^s as the supervised loss:

$$\mathcal{L}_c = \mathcal{L}_{CE}\left(y_i^s, \hat{y}_i^s\right). \tag{2}$$

Here y_i^s and $\hat{y}_i^s$ denote the i-th row of Y^s and $\hat{Y}^s$, respectively. $\mathcal{L}_{CE}$ is the cross-entropy loss.

3.4.1 Scoring Scheme. To discover the common and private classes of D^t, the scoring scheme, which estimates whether a target sample is from C, is needed.

We adopt the assumption made by [43]: target samples in C have lower uncertainty than target samples in $\overline{C}_t$. Accordingly, a well-defined scoring scheme should distinguish different degrees of uncertainty. Then we can rank the uncertainty of target samples and mark the samples with higher uncertainty as "unknown." To this end, for a sample x with the predicted label $\hat{y}$, we define a scoring scheme consisting of two signals: (1) the confidence of $\hat{y}$ (i.e., $\max(\hat{y})$). (2) the entropy of $\hat{y}$, denoted as $H(\hat{y})$.

These two signals are used to quantify the uncertainty of $\hat{y}$. On the one hand, the predictions on the source samples are more certain. Meanwhile, samples from p_C and q_C are semantically similar. Naturally, confidence is higher for samples in C and lower for samples in $\overline{C}_t$. Hence, it is reasonable to suppose that

$$\mathbb{E}_{x\sim p_C} \max(\hat{y}) > \mathbb{E}_{x\sim q_C} \max(\hat{y}) > \mathbb{E}_{x\sim q_{\overline{C}_t}} \max(\hat{y}). \tag{3}$$

On the other hand, entropy measures the smoothness of the class distribution. Similarly, we hypothesize that

$$\mathbb{E}_{x\sim q_{\overline{C}_t}} H(\hat{y}) > \mathbb{E}_{x\sim q_C} H(\hat{y}) > \mathbb{E}_{x\sim p_C} H(\hat{y}). \tag{4}$$

According to the research by [6], confidence and entropy are complementary to cover both smooth and non-smooth class distributions, which is more discriminative for uncertain predictions. Thus, the scoring scheme for a target sample x can be defined as:

$$s(x) = \max(\hat{y}) - \frac{H(\hat{y})}{\log|C_s|}, \tag{5}$$

where the entropy is normalized by its maximum value ($\log |C_s|$) so that it is restricted into [0,1]. Also, the sample scores are normalized into [0,1] in each mini-batch.

3.4.2 Separation Strategy. Based on sample scores, current UniDA methods regard samples with scores lower than a certain threshold as "unknown." Nonetheless, simply dividing target samples into common and unknown sets is sub-optimal, readily recognizing hard samples (i.e., those with scores near the threshold) as the incorrect class, leading to negative transfer [19].

As such, we additionally set a "buffer area" for hard samples, called the uncertainty set, to reduce the high risk of negative transfer during the early training. Specifically, let B_t represent the set of target nodes' indices in the mini-batch. For all nodes v_i that $i \in B_t$, $s(v_i)$ denotes the sample score corresponding to v_i. We group target samples into three sets, including the common set B_t^{co}, uncertain set B_t^{uc} and unknown set B_t^{uk}. The separation strategy is as follows:

$$i \in \begin{cases} B_t^{co}, & s(v_i) - \theta > \alpha \\ B_t^{uc}, & |s(v_i) - \theta| < \alpha \\ B_t^{uk}, & \theta - s(v_i) > \alpha \end{cases}, \tag{6}$$

where θ and α are the separation thresholds. On this basis, we calculate subsequent loss functions for different sets.

3.4.3 Prototype Contrastive Loss. To learn domain-invariant node representations in UniDA, distributions of the two domains should be aligned but limited to common classes. Consequently, we propose to use prototype contrastive learning.

The principle behind our method is to move each target point of the common set B_t^{co} to a class centroid (prototype) in D^s. Specifically, we represent the prototype from D^s belonging to class k as the mean of node representations, which can be written as

$$\mu_k^s = \frac{\sum_{i \in B_s} \mathbb{1}_{\{\text{argmax}(y_i^s)=k\}} z_i^s}{\sum_{i \in B_s} \mathbb{1}_{\{\text{argmax}(y_i^s)=k\}}}. \tag{7}$$

Here B_s is the source nodes' indices set in mini-batch, $\mathbb{1}_{\{\cdot\}}$ is an indicator function, z_i^s is the i-th node representation in Z^s. We maintain a memory bank ($\mu^s = \left[\mu_1^s, \mu_2^s, \dots \mu_K^s\right]$ with $K = |C_s|$) to store the source prototypes of each class. After that, we compute the similarity distribution vector between z_i^s and μ^s as $P_i^s = \left[P_{i,1}^s, P_{i,2}^s, \dots P_{i,K}^s\right]$, with

$$P_{i,j}^s = \frac{\exp\left(\mu_j^s \cdot z_i^s / \tau\right)}{\sum_{r=1}^K \exp\left(\mu_r^s \cdot z_i^s / \tau\right)}, \tag{8}$$

where τ is a temperature value that determines the concentration level.

To align the inter-domain distribution, we cluster the unlabeled target nodes to the prototypes. Similarly, we compute that

$$P_{i,j}^t = \frac{\exp\left(\mu_j^s \cdot z_i^t / \tau\right)}{\sum_{r=1}^K \exp\left(\mu_r^s \cdot z_i^t / \tau\right)}. \tag{9}$$

As explained by [19], pseudo-labels can be an effective tool when training a network in UniDA. Our approach is to use pseudo labels only on B_t^{co}, i.e., high-confidence target nodes that are likely to be in C. These target nodes are assigned to the confident pseudo-label $\hat{y}_i^t$, which is the i-th row of $\hat{Y}^t$. Then the inter-domain prototype contrastive loss can be written as:

$$\mathcal{L}_p = \sum_{i \in B_s} \mathcal{L}_{CE}\left(P_i^s, y_i^s\right) + \sum_{i \in B_t^{co}} \mathcal{L}_{CE}\left(P_i^t, \hat{y}_i^t\right). \tag{10}$$

3.4.4 Entropy Regularization. The prototype contrastive loss encourages target nodes from C to become well-clustered, but we still need to keep nodes from $\overline{C}_t$ away from source clusters.

Accordingly, we introduce an entropy regularization to decrease the network's confidence for samples likely to be from B_t^{uk}, i.e., low-confidence target nodes that are likely to be "unknown." For all $i \in B_t^{uk}$, the regularization can be written as

$$\mathcal{L}_e = -\sum_{i \in B_t^{uk}} H\left(\hat{y}_i^t\right). \tag{11}$$

By maximizing the entropy of $\hat{y}_i^t$, we lower the scores for "unknown" samples according to Eq. (5). That is, the regularization enforces a better separation between the scores of nodes from the private and common classes, which pushes "unknown" samples away from the decision boundary while improving the discriminability of the scoring scheme.

3.5 Intra-Domain Contrastive Alignment

Recent works on contrastive learning [14, 24, 34] show that stable and correct cluster cores can be formed by contrasting positive and negative samples in the target domain.

In light of this, we perform contrastive alignment to reduce the intra-domain discrepancy for the target domain. Specifically, we first take Z^t as anchor embedding [23] and generate semantic embedding via a target domain Multi-Layer Perceptron (MLP). This removes widely used data augmentation and discriminator from previous graph contrastive learning methods [27, 38, 46], leading to an efficient model. Then alignment of these two representations of the same node is beneficial to learn discriminative representations for classification.

3.5.1 Target Domain MLP. Due to the scarcity of labels in D^t, existing methods cannot extract discriminative node representations for G^t. In UDANE, we take advantage of the rich unlabeled data in the target domain through contrastive learning to benefit discriminative representation learning. To this end, we first adopt an MLP on X^t to generate embedding with semantic information of nodes, and this encoder is given by:

$$\tilde{Z}^{(i)} = \sigma\left(\tilde{Z}^{(i-1)} W_m^{(i)} + b_m^{(i)}\right), \tag{12}$$

where $\tilde{Z}^{(0)} = X^t$ represents the input attribute matrix of G^t, $W_m^{(i)}$ and $b_m^{(i)}$ denote the learnable weight and bias parameters, respectively. After obtaining the semantic embedding $\tilde{Z}^t$, we feed it into subsequent instance contrastive learning.

3.5.2 Instance Contrastive Loss. Afterward, we employ instance contrastive learning to learn discriminative representations for classification, minimizing the intra-domain discrepancy.

For all $i \in B_t$, given the node representations Z^t and $\tilde{Z}^t$ as the input, the instance contrastive loss is defined as:

$$\mathcal{L}_i = -\frac{1}{|B_t|}\sum_{i\in B_t}\log\frac{h\left(z_i,\tilde{z}_i\right)}{\sum_{r=1}^{|B_t|}h\left(z_i,\tilde{z}_r\right)+\sum_{r=1}^{|B_t|}\mathbb{1}_{\{r\neq i\}}h\left(\tilde{z}_i,\tilde{z}_r\right)}, \quad (13)$$

where $h\left(z_i,\tilde{z}_i\right)=\exp\left(\text{sim}\left(z_i,\tilde{z}_i\right)/\tau\right)$. We choose sim$(\cdot)$ as the dot product between a pair of normalized outputs. z_i, $\tilde{z}_i$ are representations of the i-th node in Z^t, $\tilde{Z}^t$, respectively.

By minimizing Eq. (13), we ensure that these two representations of the same node are closer to each other while the rest of the nodes in B_t are pushed apart. As a result, by ensuring consistent prediction for different representations of the same node, the decision boundary of the classifier lies in the low-density region, which is necessary for learning discriminative node representations.

Both prototype and instance contrastive losses are vital to improving model performance. On the one hand, the prototype contrastive loss reduces the inter-domain distribution discrepancy of common classes but fails to learn discriminative representations for the global target domain. On the other hand, the instance contrastive loss ensures that unlabeled target samples are consistent and are in the high-density region. However, it cannot learn the domain-invariant node representations. Hence, they are complementary to achieving universal domain adaptive representation learning.

Overall Objective. The overall objective is as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_c + \lambda\left(\mathcal{L}_p+\mathcal{L}_e+\mathcal{L}_i\right), \quad (14)$$

where λ is the same balance hyper-parameter for $\mathcal{L}_p$, $\mathcal{L}_e$ and $\mathcal{L}_i$, reducing the number of hyper-parameters.

Finally, $\mathcal{L}_c$, $\mathcal{L}_p$, $\mathcal{L}_e$ and $\mathcal{L}_i$ are jointly optimized via our objective function in Eq. (14), and all parameters are optimized using the standard backpropagation algorithms.

In the testing phase, we determine whether to categorize the sample x as "unknown" or one of the common classes based on the score $s(x)$ calculated by Eq. (5). With the separation threshold θ defined in Eq. (6), the class $y(x)$ can predicted by $\hat{y}(x)$ w.r.t. θ:

$$y(x)=\begin{cases}\text{unknown}, & s(x)<\theta\\ \text{argmax}(\hat{y}), & s(x)\geq\theta\end{cases}. \quad (15)$$

4 EXPERIMENTS

4.1 Experimental Setup

4.1.1 Datasets. Following [33], we evaluate our proposed method on three real-world attributed networks, which are widely used in cross-network node classification tasks. The statistics of the experimental datasets are shown in Table 1.

Citationv1, DBLPv7, and ACMv9 are three paper citation networks from different original sources and contain papers published in different periods. In our experiments, we consider them as undirected networks, with each edge representing a citation relation between two papers. We evaluate our proposed model on these three network domains through six cross-domain learning tasks, including C→D, A→D, D→C, A→C, D→A, and C→A, where C, D, A denote Citationv1, DBLPv7, and ACMv9, respectively.

To split categories in each dataset, we comply with the widely used rules defined in [43]. Specifically, since there are 5 classes across domains, we use the first 2 classes as the common label set C, the next 2 classes as $\overline{C}_s$, and the rest as $\overline{C}_t$.

Table 1: Statistics of the experimental datasets.

Dataset	#Nodes	#Edges	#Attributes	#Labels
Citationv1	8935	15113	6775	5
DBLPv7	5484	8130	6775	5
ACMv9	9360	15602	6775	5

4.1.2 Evaluation metrics. We use two metrics, i.e., **average class accuracy (AA)** [43] and **H-score (HS)** [6], to evaluate the performance of existing methods. The first is the mean of per-class accuracy over common and unknown classes. The second one is the harmonic mean of the accuracy on common classes (acc_{co}) and unknown class (acc_{uk}) as:

$$HS = 2\times\frac{acc_{co}\times acc_{uk}}{acc_{co}+acc_{uk}}. \quad (16)$$

In our experiments, we report the averaged results of five runs.

4.1.3 Baselines. We compare the proposed UDANE with node classification methods in single-domain (**GCN** [16] and **GraphSAGE** [13]) and cross-domain (**ACDNE** [32] and **UDA-GCN** [39]), as well as the state-of-the-art UniDA methods (**UAN** [43], **SUDA** [19] and **DANCE** [29]).

To make the baselines applicable to the universal domain adaptive node classification task, we perform the following processing: (1) For the single-domain methods, we extend them to the cross-domain scenarios by training on the source domain and testing the target domain. (2) For cross-domain node classification methods, we extend them to the UniDA setting by confidence thresholding [43]. (3) For UniDA methods, we replace their feature extractors with shared weight GCN adopted in this paper to make them adapted to networked data.

4.1.4 Implementation Details. All our experiments are performed using Pytorch [26]. The dimension of node representations for compared methods is all set to 128 for a fair comparison. In UDANE, the shared weight GCN and the target domain MLP are both constructed with two hidden layers. We use SGD with a momentum of 0.9 and a batch size of 200. The learning rate is initialed as 0.02 with the decay strategy proposed by [7]. We fixed the following hyper-parameters for all experiments: $\theta=0.5$, $\alpha=0.2$, $\tau=0.1$, λ is progressively increased from 0 to 1.

4.2 Node Classification Results

The node classification results are shown in Table 2. UDANE outperforms all the compared methods on both evaluation metrics. In particular, we have some key observations:

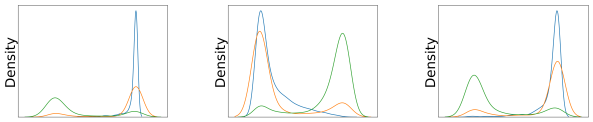
(1) The proposed UDANE model consistently beats all the baselines on six cross-domain tasks by a large margin. This verifies that the proposed UDANE can better achieve universal domain adaptation node classification by performing comprehensive information embedding, valuable knowledge transferring, and target semantics exploring in a unified, adaptive, and self-supervised learning framework.

Table 2: Node classification results (%) comparisons on six cross-domain tasks.

Methods	A→C		A→D		C→A		C→D		D→A		D→C		Average	
	AA	HS	AA	HS	AA	HS	AA	HS	AA	HS	AA	HS	AA	HS
GCN	59.25	60.57	62.27	62.29	53.27	54.80	64.86	64.96	54.17	55.79	61.14	62.90	59.16	60.22
GraphSAGE	63.43	63.78	63.48	62.81	53.38	56.41	65.02	65.57	54.50	56.62	61.43	63.21	60.21	61.40
ACDNE	47.61	46.79	44.89	45.60	50.15	49.28	52.50	51.10	50.52	51.43	51.98	52.50	49.61	49.45
UDA-GCN	50.83	49.07	47.55	48.40	51.08	50.72	57.63	55.66	50.92	51.66	49.45	48.68	51.24	50.70
UAN	55.43	56.60	54.27	56.11	60.57	62.29	59.39	61.07	59.36	60.50	68.64	68.97	59.61	60.92
SUDA	58.69	59.54	59.10	58.43	56.43	55.06	66.97	64.80	59.37	60.29	60.98	60.86	60.26	59.83
DANCE	68.87	69.67	67.67	67.04	63.32	64.89	72.15	71.58	64.19	65.22	71.88	73.08	68.01	68.58
UDANE (Ours)	**77.83**	**78.35**	**76.12**	**73.59**	**67.24**	**68.29**	**77.14**	**76.01**	**68.22**	**68.89**	**78.07**	**78.67**	**74.10**	**73.97**

Table 3: Node classification results (%) comparisons between UDANE variants on six cross-domain tasks.

Methods	A→C		A→D		C→A		C→D		D→A		D→C		Average	
	AA	HS	AA	HS	AA	HS	AA	HS	AA	HS	AA	HS	AA	HS
UDANE¬*ent*	75.69	77.26	75.46	**73.74**	67.09	68.17	76.14	75.70	67.50	68.89	76.25	78.09	73.00	73.64
UDANE¬*con*	74.62	76.65	74.52	73.16	**67.69**	**68.49**	75.48	75.55	67.66	**69.15**	76.47	78.58	72.74	73.60
UDANE¬p	73.25	72.44	69.86	67.24	63.36	63.90	72.01	69.92	63.45	64.44	71.71	72.39	68.94	68.39
UDANE¬e	68.62	70.52	70.62	71.24	61.75	63.32	70.16	70.69	61.46	62.84	68.16	69.31	66.80	67.98
UDANE¬i	74.17	74.43	72.57	70.20	64.68	65.83	73.08	72.65	65.20	66.45	72.91	74.31	70.43	70.64
UDANE	**77.83**	**78.35**	**76.12**	73.59	67.24	68.29	**77.14**	**76.01**	**68.22**	68.89	**78.07**	**78.67**	**74.10**	**73.97**



(a) confidence (b) entropy (c) final score

Figure 3: Distributions of different components of the scoring scheme on three sample groups: source samples in p_C (blue), target samples in q_C (orange) and target samples in $q_{\bar{C}_t}$ (green).

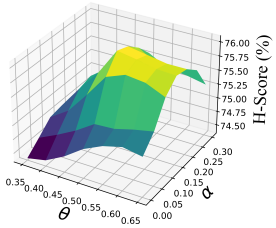
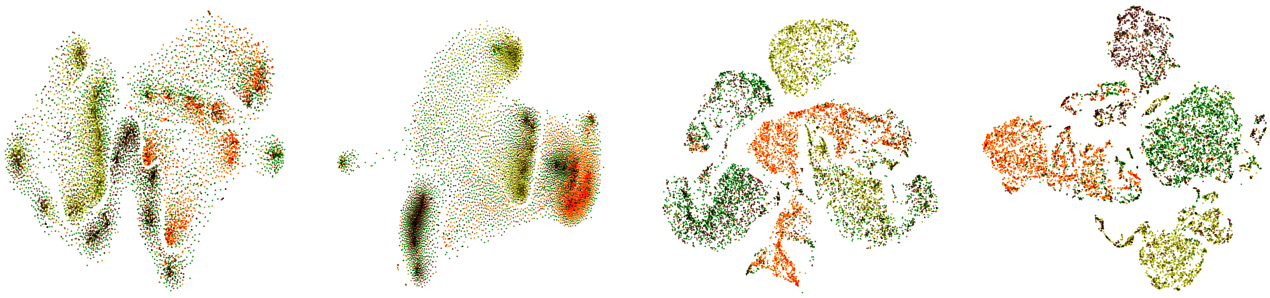


Figure 4: Classification results at different parameter settings (i.e., θ and α) from Citationv1 to DBLPv7.

(2) Existing node classification methods (GCN, GraphSAGE, ACDNE, UDA-GCN) have a worse performance among baselines. The reason is that these methods cannot deal with the varied label sets across domains. The discrepancy of label sets would lead to negative transfer, which occurs when incorrectly applying the knowledge from the source domain to the target domain, thus lowering the classification accuracy.
(3) After integrating with network embedding, existing UniDA methods (UAN, SUDA, DANCE) achieve better performance than existing node classification methods, which shows that these methods can deal with the negative transfer. However, these methods are far inferior to UDANE. The reason behind the poor performance is that these methods align the representations of the source distribution with the target distribution without considering the sample category and the target domain-specific structure, failing to generate discriminative representations.

4.3 Analysis of UDANE

4.3.1 Hypothesis Justification. To justify the validity of the hypothesis in Section 3.4.1, we plot the estimated probability density function for different components of $s(x)$ from DBLPv7 to Citationv1. As show in Figure 3(a), the hypothesis in Eq. (3) w.r.t. confidence of $\hat{y}$ is successfully verified. Similarly, we analyze the entropy of $\hat{y}$ in Figure 3(b), validating the hypothesis in Eq. (4). Finally, in Figure 3(c), we present the final sample score $s(x)$. The results show that target samples with higher scores are typically from C, explaining why the scoring scheme can correctly distinguish between target samples.



(a) GCN (b) UAN (c) ACDNE (d) UDANE

Figure 5: Visualization of the learned node representations for the target domain dataset using t-SNE [37] (The source domain is ACMv9 and the target domain is Citationv1).

4.3.2 Ablation Study. In this section, we conduct ablation studies to investigate the significance of UDANE's unique design.

First of all, to analyze the effectiveness of the scoring scheme, we compare UDANE with its variants. UDANE$\neg ent$ and UDANE$\neg con$ are the variants without integrating the entropy and confidence into the scoring scheme defined in Eq. (5), respectively. The results in Table 3 show that UDANE beats UDANE$\neg ent$ and UDANE$\neg con$ on most tasks, indicating both entropy and confidence are necessary and complementary to improve the discriminability of the scoring scheme.

One step further, we go deeper into the importance of the designed loss functions by performing an ablation study that evaluates variants of our method. UDANE$\neg p$, UDANE$\neg e$, and UDANE$\neg i$ are variants without using $\mathcal{L}_p$, $\mathcal{L}_e$ and $\mathcal{L}_i$, respectively. As shown in Table 3, the accuracies of UDANE$\neg e$ are significantly dropped, which reflects that the entropy regularization is beneficial to improve the discriminability and reliability of the scoring scheme. Also, the worse performance of UDANE$\neg p$ demonstrates that the prototype contrastive loss can effectively yield domain-invariant node representations, essential for reducing inter-domain discrepancy. Moreover, compared with UDANE, UDANE$\neg i$ verifies that the instance contrastive loss can ensure the decision boundary of the classifier lies in the low-density region, which is crucial to generate discriminative node representations.

4.3.3 Parameter Analysis. Next, we investigate the impact of θ and α in Eq. (6) with different settings. We conduct node classification on UDANE by varying θ in the range of [0.35, 0.65] and α in the range of [0, 0.3]. In this paper, we only present the visualization results from Citationv1 to DBLPv7, and similar tendencies can be observed in other settings. Results reported in Figure 4 show that our proposed UDANE consistently achieves significant performance while θ lies in the range of [0.45, 0.6] and α lies in the range of [0.15, 0.25]. In addition, UDANE's accuracies vary by about 2% w.r.t. α, verifying that our finer separation strategy mitigates the harmful effect of negative transfer.

4.4 Network Visualization

An important application of network analysis is network representation visualization [35]. In this section, we visualize the learned node representations for the target domain. To better display visualization results and perform a thorough evaluation, we evaluate the experiment under the Open-set Domain Adaptation (OSDA) [2, 20, 31] setting, a particular case of UniDA that requires no data from the source private label set. Specifically, for 5 classes across domains, we set $|C| = 3$, $\left|\bar{C}_s\right| = 0$ and $\left|\bar{C}_t\right| = 2$.

Figure 5 compares the visualization results of different representative methods, including GCN, UAN, ACDNE, and UDANE. Different colors correspond to different node labels. On the one hand, we can observe that the node representations learned by GCN and UAN lack discriminative, where the boundaries of most clusters are hard to be found and the clusters are overlapped. On the other hand, the visualization using ACDNE is not very meaningful, where many nodes belonging to the same class are separated into two clusters. For our proposed UDANE method, the target nodes are well-clustered and the decision boundaries lie in the low-density region. It confirms that the node representations learned by UDANE are sufficiently discriminative and meaningful.

5 CONCLUSION

In this paper, we study the universal domain adaptive node classification and propose a novel Universal Domain Adaptive Network Embedding (UDANE) to handle this task. Technically, we first model the network structure information in cross-network scenarios. Then we design a new scoring scheme along with a finer separation strategy to discover the common classes. On this basis, prototype-level contrastive learning and entropy regularization are used to reduce the inter-domain discrepancy adaptively. Therefore, we can learn domain-invariant node representations for facilitating valuable knowledge transfer across domains. Moreover, an instance-level contrastive learning method is incorporated to minimize the intra-domain discrepancy in a self-supervised way, learning discriminative node representations beneficial for classification. Experimental results on real-world datasets demonstrate the superior performance of UDANE over state-of-the-art algorithms and verify the effectiveness of UDANE.

ACKNOWLEDGMENTS

This work was supported by No.XDC02050200.

REFERENCES

[1] Smriti Bhagat, Graham Cormode, and S Muthukrishnan. 2011. Node classification in social networks. In *Social network data analytics*. Springer, 115–148.

[2] Pau Panareda Busto, Ahsan Iqbal, and Juergen Gall. 2018. Open set domain adaptation for image and action recognition. *IEEE transactions on pattern analysis and machine intelligence* 42, 2 (2018), 413–429.

[3] Feifei Dai, Xiaoyan Gu, Bo Li, Jinchao Zhang, Mingda Qian, and Weiping Wang. 2019. Meta-graph based attention-aware recommendation over heterogeneous information networks. In *Computational Science–ICCS 2019: 19th International Conference, Faro, Portugal, June 12–14, 2019, Proceedings, Part II 19*. Springer, 580–594.

[4] Quanyu Dai, Xiao Shen, Xiao-Ming Wu, and Dan Wang. 2019. Network transfer learning via adversarial domain adaptation with graph convolution. *arXiv preprint arXiv:1909.01541* (2019).

[5] Alex Fout, Jonathon Byrd, Basir Shariat, and Asa Ben-Hur. 2017. Protein interface prediction using graph convolutional networks. *Advances in neural information processing systems* 30 (2017).

[6] Bo Fu, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. 2020. Learning to detect open classes for universal domain adaptation. In *European Conference on Computer Vision*. Springer, 567–583.

[7] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research* 17, 1 (2016), 2096–2030.

[8] Yuan Gao, Peipeng Chen, Yue Gao, Jinpeng Wang, YoungSun Pan, and Andy J Ma. 2022. Hierarchical feature disentangling network for universal domain adaptation. *Pattern Recognition* 127 (2022), 108616.

[9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.

[10] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. 2006. A kernel method for the two-sample-problem. *Advances in neural information processing systems* 19 (2006).

[11] Andrey Gritsenko, Kimia Shayestehfard, Yuan Guo, Armin Moharrer, Jennifer Dy, and Stratis Ioannidis. 2023. Graph transfer learning. *Knowledge and Information Systems* 65, 4 (2023), 1627–1656.

[12] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 855–864.

[13] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).

[14] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9729–9738.

[15] Keith Henderson, Brian Gallagher, Tina Eliassi-Rad, Hanghang Tong, Sugato Basu, Leman Akoglu, Danai Koutra, Christos Faloutsos, and Lei Li. 2012. Rolx: structural role extraction & mining in large graphs. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1231–1239.

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

[17] Guangrui Li, Guoliang Kang, Yi Zhu, Yunchao Wei, and Yi Yang. 2021. Domain consensus clustering for universal domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9757–9766.

[18] Ziyao Li, Liang Zhang, and Guojie Song. 2019. Sepne: Bringing separability to network embedding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 4261–4268.

[19] Omri Lifshitz and Lior Wolf. 2021. Sample Selection for Universal Domain Adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 8592–8600.

[20] Hong Liu, Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Qiang Yang. 2019. Separate to adapt: Open set domain adaptation via progressive separation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2927–2936.

[21] Yijun Liu, Feifei Dai, Xiaoyan Gu, Haihui Fan, Dong Liu, Bo Li, and Weiping Wang. 2023. Powering Fine-Tuning: Learning Compatible and Class-Sensitive Representations for Domain Adaption Few-shot Relation Extraction. In *International Conference on Database Systems for Advanced Applications*. Springer, 121–131.

[22] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. 2009. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430* (2009).

[23] Yujie Mo, Liang Peng, Jie Xu, Xiaoshuang Shi, and Xiaofeng Zhu. 2022. Simple unsupervised graph representation learning. AAAI.

[24] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[25] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. IEEE Transactions on knowledge and data engineering. *22 (10)* 1345 (2009).

[26] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. (2017).

[27] Zhen Peng, Wenbing Huang, Minnan Luo, Qinghua Zheng, Yu Rong, Tingyang Xu, and Junzhou Huang. 2020. Graph representation learning via graphical mutual information maximization. In *Proceedings of The Web Conference 2020*. 259–270.

[28] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 701–710.

[29] Kuniaki Saito, Donghyun Kim, Stan Sclaroff, and Kate Saenko. 2020. Universal domain adaptation through self supervision. *Advances in neural information processing systems* 33 (2020), 16282–16292.

[30] Kuniaki Saito and Kate Saenko. 2021. Ovanet: One-vs-all network for universal domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9000–9009.

[31] Kuniaki Saito, Shohei Yamamoto, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Open set domain adaptation by backpropagation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 153–168.

[32] Xiao Shen, Quanyu Dai, Fu-lai Chung, Wei Lu, and Kup-Sze Choi. 2020. Adversarial deep network embedding for cross-network node classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 2991–2999.

[33] Xiao Shen, Quanyu Dai, Sitong Mao, Fu-lai Chung, and Kup-Sze Choi. 2020. Network together: Node classification via cross-network deep network embedding. *IEEE Transactions on Neural Networks and Learning Systems* 32, 5 (2020), 1935–1948.

[34] Ankit Singh. 2021. Clda: Contrastive learning for semi-supervised domain adaptation. *Advances in Neural Information Processing Systems* 34 (2021), 5089–5101.

[35] Jian Tang, Jingzhou Liu, Ming Zhang, and Qiaozhu Mei. 2016. Visualizing large-scale and high-dimensional data. In *Proceedings of the 25th international conference on world wide web*. 287–297.

[36] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*. 1067–1077.

[37] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[38] Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. 2019. Deep graph infomax. *ICLR (Poster)* 2, 3 (2019), 4.

[39] Man Wu, Shirui Pan, Chuan Zhou, Xiaojun Chang, and Xingquan Zhu. 2020. Unsupervised domain adaptive graph convolutional networks. In *Proceedings of The Web Conference 2020*. 1457–1467.

[40] Bencheng Yan and Chaokun Wang. 2020. Graphae: adaptive embedding across graphs. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*. IEEE, 1958–1961.

[41] Shuwen Yang, Guojie Song, Yilun Jin, and Lun Du. 2021. Domain adaptive classification on heterogeneous information networks. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*. 1410–1416.

[42] Zhilin Yang, William Cohen, and Ruslan Salakhudinov. 2016. Revisiting semi-supervised learning with graph embeddings. In *International conference on machine learning*. PMLR, 40–48.

[43] Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. 2019. Universal domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2720–2729.

[44] Yizhou Zhang, Guojie Song, Lun Du, Shuwen Yang, and Yilun Jin. 2019. Dane: Domain adaptive network embedding. *arXiv preprint arXiv:1906.00684* (2019).

[45] Ronghang Zhu and Sheng Li. 2021. Self-supervised universal domain adaptation with adaptive memory separation. In *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1547–1552.

[46] Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2021. Graph contrastive learning with adaptive augmentation. In *Proceedings of the Web Conference 2021*. 2069–2080.

[47] YuanXin Zhuang, Chuan Shi, Cheng Yang, Fuzhen Zhuang, and Yangqiu Song. 2021. Semantic-Specific Hierarchical Alignment Network for Heterogeneous Graph Adaptation. In *Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part II 21*. Springer, 335–350.