# ALEX: Towards Effective Graph Transfer Learning with Noisy Labels

**Jingyang Yuan**
National Key Laboratory for
Multimedia Information Processing,
School of Computer Science,
Peking University
Beijing, China
yuanjy@pku.edu.cn

**Xiao Luo**
Department of Computer Science,
University of California, Los Angeles
Los Angeles, USA
xiaoluo@cs.ucla.edu

**Yifang Qin**
National Key Laboratory for
Multimedia Information Processing,
School of Computer Science,
Peking University
Beijing, China
qinyifang@pku.edu.cn

**Zhengyang Mao**
National Key Laboratory for
Multimedia Information Processing,
School of Computer Science,
Peking University
Beijing, China
zhengyang.mao@stu.pku.edu.cn

**Wei Ju***
National Key Laboratory for
Multimedia Information Processing,
School of Computer Science,
Peking University
Beijing, China
juwei@pku.edu.cn

**Ming Zhang***
National Key Laboratory for
Multimedia Information Processing,
School of Computer Science,
Peking University
Beijing, China
mzhang_cs@pku.edu.cn

## ABSTRACT

Graph Neural Networks (GNNs) have garnered considerable interest due to their exceptional performance in a wide range of graph machine learning tasks. Nevertheless, the majority of GNN-based approaches have been examined using well-annotated benchmark datasets, leading to suboptimal performance in real-world graph learning scenarios. To bridge this gap, the present paper investigates the problem of graph transfer learning in the presence of label noise, which transfers knowledge from a noisy source graph to an unlabeled target graph. We introduce a novel technique termed Balance Alignment and Information-aware Examination (ALEX) to address this challenge. ALEX first employs singular value decomposition to generate different views with crucial structural semantics, which help provide robust node representations using graph contrastive learning. To mitigate both label shift and domain shift, we estimate a prior distribution to build subgraphs with balanced label distributions. Building on this foundation, an adversarial domain discriminator is incorporated for the implicit domain alignment of complex multi-modal distributions. Furthermore, we project node representations into a different space, optimizing the mutual information between the projected features and labels. Subsequently, the inconsistency of similarity structures is evaluated to identify noisy samples with potential overfitting. Comprehensive experiments on various benchmark datasets substantiate the outstanding superiority of the proposed ALEX in different settings.

*Corresponding authors.

## CCS CONCEPTS

• **Mathematics of computing** → *Graph algorithms*; • **Computer systems organization** → *Neural networks*; • **Computing methodologies** → *Learning latent representations*.

## KEYWORDS

Graph Neural Networks, Domain Adaptation, Label Noise, Graph Transfer Learning

## 1 INTRODUCTION

Graphs have gained considerable prominence in both contemporary research and practical applications, encompassing areas such as social network analysis [9, 19, 39, 66], multimedia analysis [16, 31, 48, 49], recommendation systems [3, 9, 10, 20, 47] and among others. As the development of deep learning, graph neural networks (GNNs) have emerged as a popular approach for learning effective representations of structured data [14, 21, 45], demonstrating exceptional performance across a wide array of graph machine learning challenges. The core principle of GNNs involves implementing the message passing mechanism, where each node refines its representation by obtaining and aggregating information from its neighboring nodes.

Despite their remarkable performance, current GNN-based methods are predominantly evaluated using standard benchmark datasets, which may encounter two challenges as follows when applied to real-world situations: (1) **Significant domain shift**: Contemporary methods [14, 21, 45] often focus on transductive learning within a single graph, whereas real-world applications could involve numerous online graphs [52]. However, transferring knowledge from
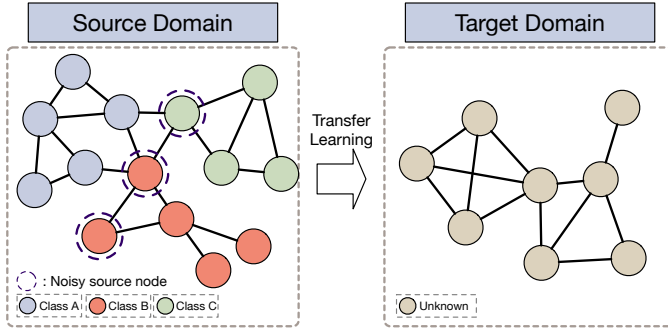
**Figure 1: The problem setting of graph transfer learning under label noise. We aim to transfer knowledge from a source graph with label noise to an unlabeled target graph.**

source graphs to target graphs remains a non-trivial problem necessitating the management of significant domain shifts. (2) **Potential noisy signals**: Typically, these methods erroneously assume that the semantic labels in the training sources are free from errors [6]. In reality, modern engineers employ machines to enhance annotation efficiency [12, 62], which could introduce label noise, leading to substantial overfitting and compromised generalization capabilities. In light of these challenges, the paper explores the practical problem of graph transfer learning under label noise, wherein GNNs are transferred from a source graph with potential label noise to an unlabeled target graph exhibiting severe domain shift.

As label noise and domain shift would have a mixed effect, it is challenging to construct a powerful GNN under these conditions. Firstly, extant graph transfer learning approaches [40, 52, 53, 65, 67] often employ explicit or implicit strategies to align data distributions across various graphs. Nevertheless, considerable label noise can impede the accurate capture of underlying multi-modal distributions [27] in graphs, rendering domain alignment arduous. Secondly, the majority of robust GNN methods are applied to semi-supervised settings wherein numerous unlabeled nodes are leveraged to supply additional semantic information [6, 32, 37]. However, a large number of nodes within the target graph could introduce potentially biased pseudo-labels due to significant domain discrepancies, leading to error accumulation. Lastly, label distributions across two graphs could exhibit considerable differences, i.e., label shift [26], which could further exacerbate the already challenging task of aligning the underlying multi-modal distributions in graphs.

In this paper, we propose a novel graph neural network named Balance Alignment and Information-aware Examination (ALEX) for this real-world problem. To mitigate the overfitting of noisy nodes, our ALEX learns robust node representations utilizing graph contrastive learning. In particular, based on the homophily assumption of graphs [46], singular value decomposition [1] is employed to generate a low-rank adjacency matrix, providing different graph views while preserving essential structural information. To surmount domain shift along with label shift, we estimate a prior distribution that generates subgraphs with balanced label distributions. Subsequently, we construct a domain discriminator conditioned on both node embeddings and label distributions, which are trained to align underlying multi-modal distributions behind graphs in

an adversarial manner. Additionally, we identify potentially noisy nodes based on mutual information. Specifically, we project node representations into a different space in which the mutual information between their representations and labels is maximized. As graph structures are not incorporated, the embedding vectors of noisy nodes could be substantially influenced. Therefore, in order to eliminate label noise, we select nodes exhibiting high inconsistency in similarity structures for each node between the two spaces. To summarize, the contribution of this paper is as follows:

- **Problem:** This paper studies an underexplored yet realistic problem of graph transfer learning under label noise and proposes a unified method ALEX to tackle this.
- **Methodology:** Our ALEX not only studies both prior distributions for balanced domain alignment of robust node representations but also estimates the inconsistency of similarity structures to detect noisy nodes after maximizing mutual information between representations and labels.
- **Evaluation:** Extensive experiments on various datasets validate the superiority of ALEX compared with state-of-the-art baselines.

## 2 RELATED WORK

### 2.1 Graph Neural Networks

Due to their high capacity for modeling graph-structured data, graph neural networks (GNNs) are gaining popularity in many communities [9, 18, 28, 30, 39, 66]. After simplification from spectral methods, the majority of graph neural networks adhere to the message passing principle [21], where each node updates its representation by iteratively aggregating information from neighbors. There have been extensive efforts to enhance the effectiveness and efficacy of GNNs from multiple perspectives, including architecture design [55, 56], trustworthy [50, 59] and invariance theory [5, 54]. For instance, GraphSAGE [14] accelerates the rate at which messages travel through sampling nodes in the vicinity surrounding each node. CIGA [5] employs causal models to characterize the generation of graphs and enhances the generalization ability by invariant learning. However, these methods do not account for realistic scenarios involving label noise and domain shift, which could impact their practice use. Towards this end achieve this objective, we proposes a novel method ALEX for these realistic scenarios.

### 2.2 Domain Adaptation

As a popular topic in the multimedia community, domain adaptation seeks to convey semantic information from a labeled source domain to an unlabeled target domain [23, 57, 64]. Earlier attempts measured the distance between the source and target domains using different matrices, e.g., maximum mean discrepancy [4] (MMD) and enhanced transport distance [24] (ETD), which would be minimized for explicit domain alignment. An alternative solution to the problem is to introduce a domain discriminator that is trained adversarially for implicit domain alignment [17, 25, 44, 63]. Domain adaptation has also been introduced in multiple investigations in graph data mining [40, 52, 65]. For instance, UDA-GCN [52] employs local and global graph neural networks to discover similarity relationships and a domain classifier to align node embeddings on two graphs. AdaGCN [7] leverages both semi-supervised learning and adversarial learning techniques to improve performance. Graph

Contrastive Learning Network [53] (GCLN) introduces graph contrastive learning to enhance transfer learning on graphs, whereby similarity relationships are learned from the perspectives of attraction and repulsion forces. In contrast, our ALEX estimates prior distributions and sample subgraphs to surmount potential label shifts, which guide the reliable learning of node representations.

## 2.3 Learning with Label Noise

Label noise would inevitably be introduced during large-scale data annotation [41, 61]. Consequently, learning with label noise has received much attention recently and has been applied to various problems, including image segmentation [33] and entity alignment [35]. Potential solutions include modifying loss objectives [15, 22, 34, 38] by incorporating the noise transition matrix or selecting noisy samples with a high degree of confidence. For example, Co-Teaching [15] introduces two deep neural networks that are independently trained and provide each other with communications about trustworthy examples. A small number of studies on learning with label noise on graphs have been conducted recently [6, 32, 37, 60]. NRGNN [6] explores unlabeled nodes alongside labeled nodes to provide more semantic information. PI-GNN [8] concentrates on learning pairwise supervised signals based on graph structure, thereby eliminating potential overfitting in graph neural networks. In this paper, we train an extra projector by maximizing mutual information between node representation and labels, then we choose nodes with high inconsistency similarity structures as noisy ones.

## 3 METHODOLOGY

### 3.1 Framework Overview

This paper investigates the problem of graph transfer learning under label noise, which is challenging due to the combined influence of domain shift and label noise. Here, we propose a novel method named ALEX, which is featured three primary components:

- **Robust Representation Learning.** Considering that label noise and domain shift can introduce potential biases, we employ graph contrastive learning based on singular value decomposition to produce robust representations on two graphs.
- **Balanced Domain Alignment.** To effectively address domain shift alongside prospective label shifts between source and target graphs, we compute a prior distribution to generate subgraphs with balanced label distributions. Subsequently, we apply a domain discriminator to facilitate the alignment of underlying multi-modal distributions in an adversarial manner.
- **Mutual Information-aware Refinement.** We transform node embeddings into an alternative space that maximizes mutual information with label information and identify nodes exhibiting high inconsistency in similarity structures between the two embedding spaces as potentially noisy ones.

### 3.2 Problem Definition

Given a labeled source graph consisting of $N^s$ nodes, denoted as $\mathcal{G}^s = (\mathcal{V}^s, \mathcal{E}^s)$, and an unlabeled target graph comprising $N^t$ nodes, denoted as $\mathcal{G}^t = (\mathcal{V}^t, \mathcal{E}^t)$. The attribute matrices for the two graphs are represented by $X^s \in \mathbb{R}^{N^s \times F}$ and $X^t \in \mathbb{R}^{N^t \times F}$, respectively, where $F$ indicates the dimension of node attributes. The adjacency

matrices for the two graphs are denoted as $A^s \in \{0, 1\}^{N^s \times N^s}$ and $A^t \in \{0, 1\}^{N^t \times N^t}$, respectively. The label for each source node $i^s \in \mathcal{G}^s$ is represented by $y_i^s \in \{1, 2, \cdots, C\}$, where $C$ denotes the number of classes and may contain a certain degree of noise. Our objective is to predict the label $y_i^t$ for each node on the target graph $i^t \in \mathcal{G}^t$ with potential domain shifts.

### 3.3 Robust Representation Learning Using Singular Value Decomposition

To eliminate the confounding effects of domain shift and label noise, we introduce an unsupervised component that generates robust node representations without data annotations for every graph. Here, a basic assumption for homophily graphs [46] is that nodes with similar labels and features tend to be connected, indicating a low rank of similarity matrix based on the ground truth. Therefore, we generate different views using singular value decomposition [1] (SVD) on the adjacency matrix. These views are compared with original graphs to formulate contrastive loss, which helps to generate robust representations devoid of data annotation.

In particular, we first normalize the adjacent matrix and then introduce the SVD algorithm as in [13], which generates an orthonormal matrix with rank $q$. Formally,

$$U^s, S^s, V^{s\top} = \text{SVD}(D^{s^{-\frac{1}{2}}} A^s D^{s^{-\frac{1}{2}}}, q), \tag{1}$$

$$\hat{A}^s = U^s S^s V^{s,\top}, \tag{2}$$

where $D^s$ is the degree matrix of $A^s$ for normalization. $U^s$, $S^s$ and $V^{s,\top}$ are three $p$-rank matrix from the approximate SVD algorithm [13]. Then, we leverage graph convolution layers $GC_\theta^l$ with parameters $\theta$ to update node representations using our augmented adjacency. In formulation, we have:

$$\hat{H}^{s,(l)} = \text{GC}_\theta\left(\hat{A}^s, \hat{H}^{s,(l-1)}\right) \tag{3}$$

where $\hat{H}^{s,(l)} \in \mathbb{R}^{N^s \times d}$ is the augmented hidden matrix for source graph at layer $l$, $W^{(l)}$ is the weight matrix in graph convolution module and $\sigma(\cdot)$ is an activation function. After stacking $L$ layers, the final node embeddings are written as:

$$\hat{H}^s = [\hat{h}_1^s, \cdots, \hat{h}_N^s] = GC_\theta(\hat{A}^s, X^s), \tag{4}$$

where $\hat{h}_i^s$ is the embedding for node $i^s \in \mathcal{G}^s$. Note that $GC_\theta$ can be implemented using vanilla graph convolution [21] or more flexibly by adopting sophisticated architectures [51, 52].

Subsequently, we compare the node embeddings from augmented graphs with those obtained from original graphs, following the principle of graph contrastive learning. Specifically, we produce an original node embedding matrix as:

$$H^s = \text{GC}_\theta\left(D^{s^{-\frac{1}{2}}} A^s D^{s^{-\frac{1}{2}}}, X^s\right) \tag{5}$$

where every weight matrix $W^{(l)}$ is shared for both node embedding matrices. Finally, a graph contrastive learning objective is utilized to generate robust representations as follows:

$$\mathcal{L}_{CL}^s = -\sum_{i=1}^{N^s} \log \frac{\exp(h_i^s \star \hat{h}_i^s / \tau)}{\sum_{j=1}^{N^s} \exp(h_i^s \star \hat{h}_j^s / \tau)}, \tag{6}$$
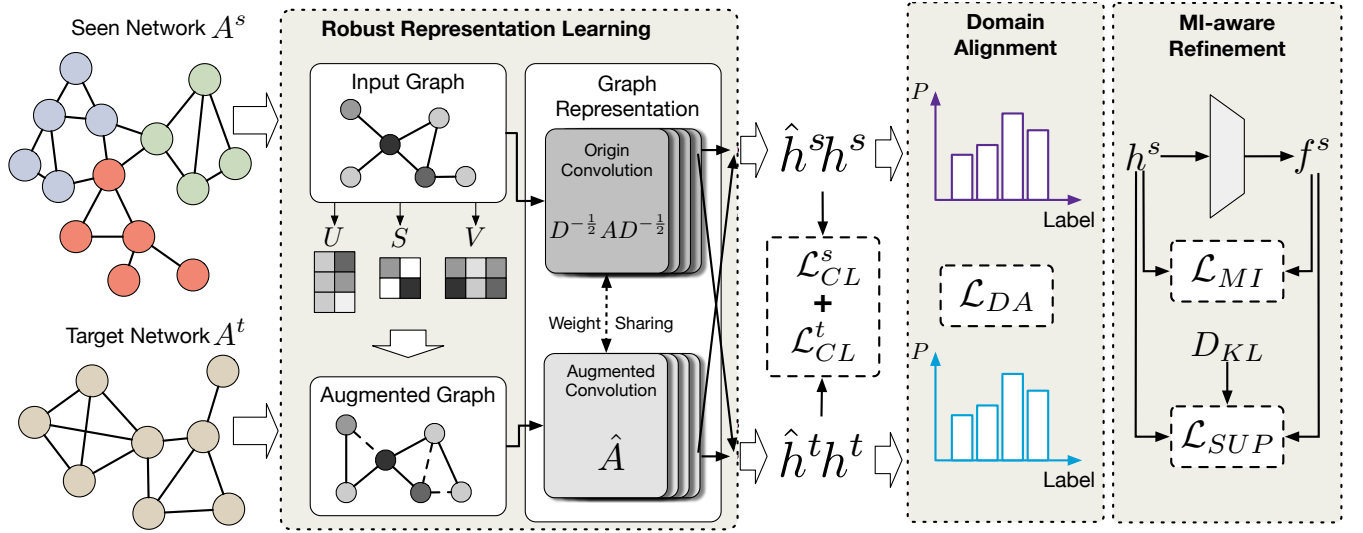
Figure 2: Overview of our proposed ALEX. Our ALEX first generates augmented graphs using singular value decomposition and then compares node representations across views with contrastive learning. Moreover, ALEX computes prior label distributions for balanced domain alignment. To clean label noise, ALEX maps node representations into a different space and then measures similarity structures to identify noisy nodes.

where $\star$ calculates the cosine similarity of two vectors and $\tau$ is a temperature parameter set to 0.5 suggested by [29]. Similarly, we generate a contrastive objective for node embeddings in the target graph. Then the node embeddings for original and augmented target graphs are represented as $\boldsymbol{h}_i^t$ and $\hat{\boldsymbol{h}}_i^t$, which results in $\mathcal{L}_{CL}^t$. The final objective for obtaining robust representations is as follows:

$$\mathcal{L}_{CL} = \mathcal{L}_{CL}^s + \mathcal{L}_{CL}^t \qquad (7)$$

This component does not conduct node representation learning on two single graphs without label information, which can potentially generate robust representations in our scenarios with significant label noise and domain discrepancy issues.

### 3.4 Subgraph Sampling for Balanced Domain Alignment

Existing methods employ an adversarial domain discriminator [25, 52] to guide domain alignment to address domain discrepancy between source and target graphs. However, potential label shift [26] between these two graphs is ignored. In other words, the proportion of each class could vary between the source and target graphs, making it difficult to align underlying multi-modal distributions [27]. Therefore, we propose subgraph sampling, which calculates the prior distribution for each node and then sample subgraphs for balanced label distribution. Then, both hidden features and label distributions are taken into account to identify complex underlying structures for domain alignment.

In particular, we first introduce an MLP classifier $f_\theta(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^C$, which maps hidden embedding to label space. The classifier is well-trained by the source graph with the cross-entropy objective:

$$\mathcal{L}_{SUP} = -\frac{1}{N^s} \sum_{i^s \in \mathcal{G}^s} \boldsymbol{y}_i^{sT} \log \boldsymbol{p}_i^s, \qquad (8)$$

where $\boldsymbol{p}_i^t = f_\theta(\boldsymbol{h}_i^t)$ is the predicted distribution and $\boldsymbol{y}_i^s$ is the one-hot label embedding. Then, a pseudo-label can be generated for every target node, i.e., $\hat{y}_i^t = \arg\max_c \boldsymbol{p}_i^t[c]$. Then, we can estimate the label distribution of the two graphs, i.e., $p(y)$. In formulation,

$$p(y)[m] = \frac{\#\{i|y_i^s = m\} + \#\{j|\hat{y}_j^t = m\}}{N^s + N^t}, \qquad (9)$$

where $p(y)[m]$ denotes the probability of class $m$ in the dataset. Then, we sample nodes from both source and target graphs based on $p(y)$, which generates subgraphs $\tilde{\mathcal{G}}^s$ and $\tilde{\mathcal{G}}^t$ with balanced label distributions. To achieve domain alignment for underlying multi-modal distributions, a domain discriminator conditioned on both hidden features and labels is introduced, which guides feature learning in an adversarial manner. In formulation, we utilize an MLP to formalize $D_\phi$ and the objective is stated as:

$$\mathcal{L}_{DA}(\theta, \phi) = -\sum_{i \in \hat{\mathcal{G}}^s} \log D_\phi([\boldsymbol{h}_i^s, \boldsymbol{y}_i^s]) - \sum_{j \in \hat{\mathcal{G}}^t} \log(1 - D_\phi([\boldsymbol{h}_j^t, \boldsymbol{p}_j^t])), \qquad (10)$$

where hidden information and label information is concatenated as the input of $D_\phi$. On the one side, the objective is minimized with regard to $\phi$ to distinguish two domains utilizing all available resources. On the other side, the objective is maximized with respect to $\theta$ to confound the domain discriminator, thereby decreasing the domain disparity of multi-modal distributions in the hidden space. By employing subgraph sampling, the influence of label shifts is mitigated, reducing bias when aligning multimodal distributions.

### 3.5 Mutual information-aware Refinement

Nevertheless, label noise could still be detrimental to the optimization procedure. One potential solution is to identify noisy samples [15, 22]. Existing methods concentrate on independent and

---

**Algorithm 1** Training Algorithm of ALEX

---

**Input:** Source graph $\mathcal{G}^s$; Target graph $\mathcal{G}^t$; Percentile $\alpha$ for label refinement; SVD order $q$;

**Output:** Parameters $\theta$ and $\phi$;

1: Warm up the network by minimize $\mathcal{L}_{SUP} + \mathcal{L}_{CL}$;
2: Generate the augmented adjacent matrix using Eqn. 2;
3: **for** $c = 1, 2, \cdots, C$ **do**
4:     Obtain the clean node set using Eqn. 15;
5:     Calculate $p(y)$ using Eqn. 9;
6:     Generate subgraph $\mathcal{G}^s$ and $\mathcal{G}^t$ based on $p(y)$;
7:     **repeat**
8:         Generate hidden embeddings for both original and augmented graphs using Eqns. 3 and 5, respectively.
9:         Update parameters of $G(\cdot)$ and $D(\cdot)$ using Eq. 17;
10:     **until** Convergence
11: **end for**

---

identically distributed (i.i.d) data [15, 22, 38], which makes their applications to graph data challenging. To this end, we introduce a mutual information-based refinement module that projects node representations to another embedding space by maximizing mutual information with labels. Since graph structures are not involved, noisy nodes can be significantly influenced. Therefore, nodes with high inconsistency of similarity structures between two spaces are subsequently considered to be noisy.

In particular, we first introduce an MLP $g_\phi(\cdot)$, which maps node embeddings into another space, i.e., $f_i^s = g_\phi(h_i^s)$. Here, the mutual information between $f_i^s$ and $y_i^s$ is maximized. Afterwards, we follow MINE [2] to calculate the lower bound of $I(f_i^s, y_i^s)$ by introducing an MI estimator $T$. Then, the objective can be formulated as:

$$\mathcal{L}_{MI} = \frac{1}{N^s} \sum_{i=1}^{N^s} \left[ T(f_i^s, y_i^s) \right] - \log\left( \frac{1}{N^{s^2}} \sum_{i=1}^{N^s} \sum_{j=1}^{N^s} \left[ e^{T(f_i^s, y_j^s)} \right] \right). \quad (11)$$

Since the absolute positions of all these deep features would change after the projection, we compare the similarity structures rather than positions to identify noisy nodes. Here, we define similarity structures by introducing a range of anchor nodes, i.e., $\mathbb{S} = \{i_1^s, \cdots, i_K^s\} \in \mathcal{G}^s$ and the similarity distribution between each node and anchor nodes in the hidden space can be formulated as:

$$w_i^s[k] = \frac{\exp\left( h_i^s \star h_{i_k}^s / \tau \right)}{\sum_{k'=1}^{K} \exp\left( h_i^s \star h_{i_{k'}}^s / \tau \right)}. \quad (12)$$

Similarly, the similarity distribution within the newly established embedding space can be described as follows:

$$r_i^s[k] = \frac{\exp\left( f_i^s \star f_{i_k}^s / \tau \right)}{\sum_{k'=1}^{K} \exp\left( f_i^s \star f_{i_{k'}}^s / \tau \right)}, \quad (13)$$

Subsequently, we measure the KL divergence between the two distributions obtained above as inconsistency scores:

$$d_i = D_{\mathrm{KL}}(w_i^s \| r_i^s) \quad (14)$$

Given a threshold $\mu$, we identify samples with large inconsistency scores and systematically eliminate them in supervised training.

**Table 1: Dataset Statistics. A, C, and D represent ACM, Citation, and DBLP, respectively. The number in the front of dataset name indicates the specific group. Cross domain tasks are assigned within the same group.**

| Dataset | # Label | # Feature | # Node | # Edge | Origin Data |
|---|---|---|---|---|---|
| 1A | 6 | 7537 | 5578 | 7341 | ACMv9 |
| 1D | 6 | 7537 | 11135 | 7410 | DBLPv8 |
| 2A | 5 | 6775 | 9369 | 15602 | ACMv9 |
| 2C | 5 | 6775 | 8935 | 15113 | Citationv1 |
| 2D | 5 | 6775 | 5484 | 8130 | DBLPv7 |

The clean node set is written as follows:

$$C = \{i^s \in \mathcal{G}^s | d_i < \mu(\alpha)\}. \quad (15)$$

where $\mu(\alpha)$ is determined by the $\alpha$ percentile threshold of divergence in the whole source graph. In the end, we revise the supervised learning loss in Eqn. 8 as follows:

$$\mathcal{L}_{SUP} = -\frac{1}{|C|} \sum_{i^s \in C} y_i^{sT} \log p_i^s. \quad (16)$$

## 3.6 Framework Summarization

In a nutshell, the whole loss objective for our proposed ALEX can be succinctly expressed as follows:

$$\min_\theta \max_\phi \mathcal{L} = \mathcal{L}_{SUP} + \mathcal{L}_{CL} - \mathcal{L}_{DA} + \mathcal{L}_{MI}. \quad (17)$$

where $\phi$ is the parameters of $D(\cdot)$ and $\theta$ is for graph neural network. Here, we fix $\phi$ to minimize $\mathcal{L}_{DA}$ and then minimize $\mathcal{L}$ with respect to $\theta$ with fixed $\phi$, which follows the paradigm of adversarial learning. We first warm up our ALEX using contrastive learning loss and supervised loss. The algorithm is summarized in Algorithm 1.

## 4 EXPERIMENT

In this section, we showcase the experimental findings to establish the potency of ALEX. To start, we will provide an overview of the datasets, baselines, and configurations, followed by a presentation of the results and a comprehensive analysis. We aim to answer the following five research questions in detail.

**RQ1:** How does the proposed ALEX perform on real-world datasets compared with current state-of-art methods?

**RQ2:** Is our ALEX robust against different levels of label noise?

**RQ3:** How do the hyper-parameters in ALEX affect the cross-domain classification performance ?

**RQ4:** What is the role of the main components in our proposed ALEX and how do they impact the overall performance?

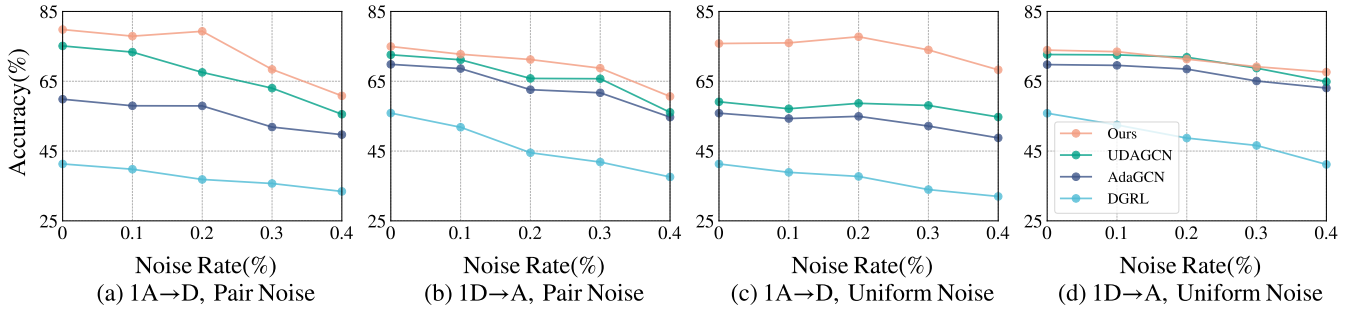**RQ5:** Is there any additional analysis that can confirm the superiority of ALEX compared to baseline models?

### 4.1 Experimental Setup

**Benchmark Datasets.** In this experiment, we diligently assess the performance of our proposed model by examining it on three real-world networks sourced from ArnetMiner [43]. The networks are constructed based on paper citation information obtained from

**Table 2: Summary of accuracy on eight cross-domain classification tasks with Pair Noise. The best performance is highlighted in boldface. Our proposed method ALEX outperforms all the baseline methods in most cases.**

| Dataset | MLP | DeepWalk | GraphSAGE | LINE | GCN | DGRL | AdaGCN | UDAGCN | Ours |
|---------|-----|----------|-----------|------|-----|------|--------|--------|------|
| 1A→D | 38.12±1.9 | 28.93±3.2 | 35.60±9.6 | 35.20±1.1 | 56.88±3.2 | 36.87±0.9 | 57.94±5.8 | 67.54±11.2 | **79.31±6.4** |
| 1D→A | 48.71±2.9 | 37.44±1.5 | 36.52±2.2 | 35.09±2.6 | 65.60±1.1 | 44.52±4.8 | 62.59±2.8 | 65.81±2.7 | **71.22±1.4** |
| 2A→D | 43.11±1.4 | 15.98±3.9 | 52.96±3.4 | 18.57±2.1 | 52.10±2.4 | 40.32±1.6 | 51.23±6.1 | 52.96±7.7 | **54.54±4.4** |
| 2D→A | 34.90±2.0 | 30.28±2.4 | 36.42±4.3 | 31.67±1.8 | 34.08±2.3 | 32.86±3.8 | 36.93±5.4 | 37.40±7.2 | **56.39±4.2** |
| 2A→C | 39.58±1.7 | 23.19±5.8 | 48.89±5.7 | 24.64±5.7 | 46.30±5.6 | 36.26±2.4 | 47.01±7.5 | 47.11±9.1 | **64.88±6.0** |
| 2C→A | 38.69±2.0 | 19.43±3.9 | 45.77±5.0 | 22.54±5.3 | 47.10±4.4 | 35.34±2.3 | 43.65±4.7 | 43.36±6.8 | **60.79±4.7** |
| 2C→D | 44.74±1.7 | 13.29±1.7 | 57.26±2.0 | 15.18±1.6 | 57.69±2.7 | 41.95±1.0 | 55.63±4.4 | **57.20±6.1** | 56.96±3.7 |
| 2D→C | 35.32±2.3 | 24.36±5.1 | 38.35±6.2 | 24.76±1.1 | 35.67±4.5 | 37.46±2.5 | 41.36±9.1 | 41.58±11.3 | **59.93±4.1** |



Figure 3: Accuracy on 1A→D and 1D→A with Pair Noise and Uniform Noise on various levels of label noise setting.

various databases, namely DBLP, ACM, and Microsoft Academic Graph. Detailed dataset information is presented in Table 1.

Since all labels in the datasets are clean, following [6, 34, 58], we employ two different disturbance methods to add noise to the training labels of the original datasets. (i) Uniform Noise: The label of each node is randomly switched to another class with a certain probability. (ii) Pair Noise: The label of each node is changed exclusively to its most similar class with a certain probability. In both two methods, the probability is a parameter set to 20% by default. To appropriately choose the best model and assess its performance, we split the dataset into training, testing, and validation sets, maintaining a 7:2:1 proportion for each respective set.

**Evaluation Protocols.** To evaluate our algorithm for cross-domain node classification, we utilized two distinct sets of data partitioning. The first set comprises DBLPv7, ACMv9, and Citationv1, which have been commonly used in previous domain adaptation studies [7]. The second set consists of more recent data, DBLPv8, and ACMv9, with slightly different feature preprocessing [52].

To simplify the notation, we referred to the dataset groups using their corresponding numbers and initials. Specifically, we designated the group with DBLPv8 and ACMv9 as Group 1, and the older group with DBLPv7, ACMv9, and Citationv1 as Group 2. Therefore, we used the notation 2C to refer to Citationv1 from Group 2. We conducted multi-label classification on these three network domains, performing eight transfer learning tasks: 1A→D, 1D→A, 2A→D, 2D→A, 2A→C, 2C→A, 2C→D, and 2D→C.

We assess all models by conducting a grid search within the hyperparameter space and subsequently present the optimal results

for each respective approach. For each task, we perform ten trials with all models by varying the random seed and report the accuracy in percentage with standard deviation for comparison.

**Baseline Methods.** We compare ALEX with state-of-the-art models. Note that for fairly evaluating our model and highlight its performance, our approach is tested against both advanced single-domain node classification models and cross-domain models. The single-domain models consist of methods that ignore graph information (MLP), models that use neighborhood information and graph structure (DeepWalk [36], LINE [42]), and graph neural network models (GraphSAGE [14], GCN [21]). The cross-domain models include DGRL [11], AdaGCN [7], and UDAGCN [52], which employs gradient reverse layer (GRL) for domain classification. DGRL utilizes an MLP as a feature generator while AdaGCN and UDAGCN adopt GCN architecture to focus on graph representation learning.

## 4.2 Performance Comparison (RQ1)

Table 2 and Table 3 illustrate the accuracy in percentage with standard deviations of ALEX and baseline models across all tasks, considering the Pair and Uniform Noise with default noise rate of 20%. The results reveal the following insights:

- Models employing traditional encoders, such as MLP and DGRL, show suboptimal performance due to their inability to capture the complex relationships within graph structure. This underscores the necessity of leveraging the inductive bias of graph structures.
- Graph-based approaches, such as DeepWalk, LINE, GraphSAGE, and GCN, demonstrate enhanced performance in comparison to traditional methods, highlighting the benefits of leveraging both

**Table 3: Summary of accuracy on eight cross-domain classification tasks with Uniform Noise. The best performance is highlighted in boldface. Our proposed method ALEX outperforms all the baseline methods in most cases.**

| Dataset | MLP | DeepWalk | GraphSAGE | LINE | GCN | DGRL | AdaGCN | UDAGCN | Ours |
|---|---|---|---|---|---|---|---|---|---|
| 1A→D | 37.77±1.7 | 27.19±3.8 | 30.22±4.1 | 32.83±1.7 | 49.32±3.8 | 37.73±1.4 | 54.95±7.2 | 58.68±9.9 | **77.75±5.3** |
| 1D→A | 50.56±3.0 | 38.70±1.2 | 43.92±4.8 | 36.05±2.0 | 68.29±1.3 | 48.72±2.8 | 68.48±1.7 | **71.88±1.9** | 71.33±2.4 |
| 2A→D | 45.25±1.7 | 13.68±4.9 | 55.18±4.8 | 14.82±2.0 | 41.88±6.1 | 41.37±0.7 | 48.14±7.1 | 48.05±9.1 | **63.12±4.4** |
| 2D→A | 35.31±2.0 | 31.93±2.4 | 36.76±3.5 | 32.44±1.9 | 31.58±2.7 | 34.12±2.8 | 35.26±7.5 | 33.34±8.6 | **62.63±2.5** |
| 2A→C | 40.54±1.7 | 26.96±5.9 | 50.93±5.7 | 28.02±5.8 | 33.62±5.9 | 40.89±2.3 | 39.77±7.1 | 42.03±9.8 | **68.63±4.6** |
| 2C→A | 39.98±3.3 | 17.86±5.1 | 49.59±7.3 | 24.08±2.4 | 38.73±8.2 | 39.98±1.6 | 43.84±4.4 | 42.10±6.7 | **65.81±3.3** |
| 2C→D | 47.10±4.7 | 11.62±1.3 | 62.14±4.6 | 13.49±1.3 | 53.98±2.6 | 43.71±1.5 | 56.74±5.5 | 55.45±9.9 | **66.73±4.1** |
| 2D→C | 36.08±2.5 | 24.64±6.1 | 38.98±3.6 | 26.47±0.9 | 29.51±4.4 | 36.08±1.9 | 39.58±4.4 | 35.46±10.6 | **68.18±4.3** |



(a) $q$ Impact on 1A→D  (b) $q$ Impact on 1D→A  (c) $\alpha$ Impact on 1A→D  (d) $\alpha$ Impact on 1D→A
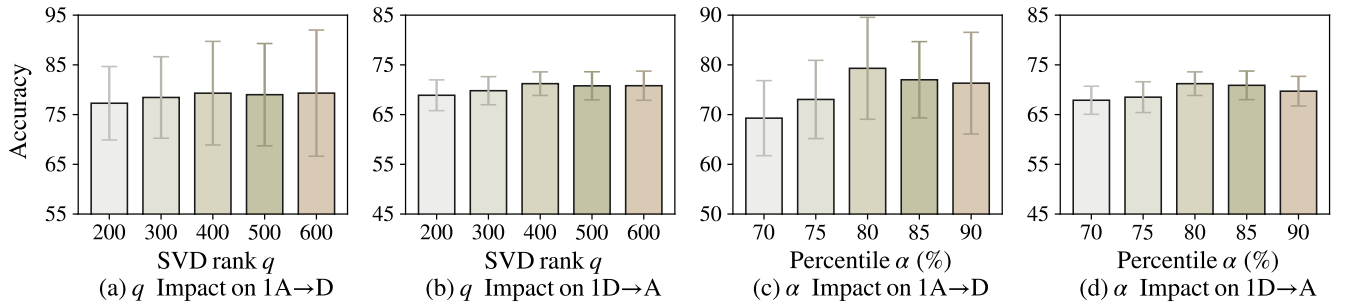
**Figure 4: Parameter Sensitivity Analysis on 1A→D and 1D→A tasks with Pair Noise. Bar charts and error bars depict accuracy and corresponding 80% confidence intervals. (a)(b) display performance on different SVD rank $q$ and (c)(d) shows performance varying percentile $\alpha$ of KL divergence threshold.**

local graph structures and node features for superior representation. Furthermore, GraphSAGE and GCN stand out due to their graph convolution capabilities, which allow models to preserve structural features on multiple scales.

- Multi-task learning models, such as DGRL and AdaGCN, outperform single-task models, indicating the benefits of jointly learning from multiple tasks. Nevertheless, these models still underperform compared to our proposed ALEX, as they are notably impacted by noisy labels. This underscores the challenge of multitask node classification in the presence of label noise.
- Our proposed ALEX consistently outperforms all baseline models across the majority of tasks, showcasing its exceptional ability to capture the complex relationships within multi-domain graph-structured data and reduce the impact of noise. Specifically, ALEX exceeds the best-performing baseline by 28.60% and 25.87% on 2D→C and 2D→A, respectively. Additionally, ALEX displays a relatively smaller standard deviation compared to other graph-based multi-task models, signifying a more robust and stable performance across diverse training scenarios.

### 4.3 Effects of Different Noisy Rates (RQ2)

To demonstrate the effectiveness of the ALEX model in handling different levels of label noise, we evaluate its performance with noise rates ranging from {0, 10, 20, 30, 40}%. We also compare our method with competitive baselines. We evaluate our method and baselines on the tasks of 1A→D and 1D→A with both Pair and Uniform

Noise. The mean performance across ten trials is demonstrated in Figure 3. Our observations from the figure are as follows:

- Even in the presence of low or higher label noise, the proposed ALEX surpasses the other models. This is attributed to the robust representation learning of Singular Value Decomposition based contrastive learning, which enhances GNN message-passing process. Also, balanced domain alignment contributes to performance by reducing classification bias with or without noise.
- With the increase in label noise levels, a significant decline in the performance of all baselines is observed. Although the performance of ALEX is also affected, it still maintains a lead over the baselines, which exhibits greater resilience in the face of label noise, indicating the efficacy of the proposed method in managing noisy labels by mutual information-aware refinement.

### 4.4 Parameter Sensitivity Analysis (RQ3)

In this section, we study the impact of two hyper-parameters in our ALEX: the SVD rank $q$ in the SVD-based graph contrastive learning module and the percentile threshold $\alpha$ in the mutual information-aware refinement procedure, which measures whether inconsistency scores of each node are significant enough to be retained in supervised training. We evaluate the model's performance with different values of $q$ and $\alpha$ and report the results on a chosen dataset. **Impact of SVD rank $q$:** We conducted experiments by varying the SVD rank $q$ in the SVD-based graph contrastive learning module from 200 to 600 and evaluated the performance on transfer learning

Jingyang Yuan et al.

**Table 4: Comparisons between our ALEX and its variants in different settings with Pair Noise.**

| Dataset | ALEX w/o G | ALEX w/o B | ALEX w/o M | ALEX |
|---------|-----------|-----------|-----------|------|
| 2A→D | 49.44±9.5 | 50.57±5.1 | 53.14±3.7 | **54.54±4.4** |
| 2D→A | 40.83±8.3 | 38.91±4.6 | 54.07±3.9 | **56.39±4.2** |
| 2A→C | 47.46±9.3 | 47.25±6.3 | 56.73±5.0 | **64.88±6.0** |
| 2C→A | 49.00±6.1 | 45.29±5.0 | 49.72±3.2 | **60.79±4.7** |
| 2C→D | 55.21±8.9 | 56.16±3.7 | **57.36±3.9** | 56.96±3.7 |
| 2D→C | 47.80±10.8 | 44.00±3.8 | 57.92±4.2 | **59.93±4.1** |

tasks 1A→D and 1D→A. The results are presented in Figure 4. We observe that the performance is relatively stable across different values of $q$, which indicates that our ALEX model is robust to changes in the SVD rank $q$. However, the variations in performance may increase with a larger value of $q$. Additionally, it can be noticed that the best performance was achieved when setting $q$ to 400. So we kept $q = 400$ by default in other experiments.

**Impact of $\alpha$**: We conducted experiments to investigate the impact of the threshold in the mutual information-aware refinement module. We varied the percentile of threshold, $\alpha$, over $\{70, 75, 80, 85, 90\}\%$ and plotted the results in Figure 4. It can be observed that the classification performance improves gradually when $\alpha$ increases from 70% to 80%. This improvement is mainly due to the fact that lower threshold values set a more strict criterion for inclusion, leading to the removal of more training samples. However, we also found that excessively large threshold values could have a negative impact on performance, as they tend to increase the number of noisy samples that interfere with the optimization procedure. So it seems that the best $\alpha$ value should be set between 75% and 85%. Our experiment suggests that this range provides optimal results in different tasks.

## 4.5 Ablation Study (RQ4)

To understand the effectiveness of each component in our proposed ALEX, we conduct an ablation study by comparing different variants of ALEX. We investigate the impact of the SVD-based graph contrastive learning, balanced domain alignment module, and mutual information-aware refinement procedure for denoising. The results of the ablation study are presented in Table 4, and the following ALEX variants are designed for comparison:

- ALEX w/o G: A variant of ALEX with the SVD-based graph contrastive learning component removed.
- ALEX w/o B: A variant of ALEX with the subgraph sampling for the balanced domain alignment module removed.
- ALEX w/o M: A variant of ALEX with the mutual information-aware refinement procedure removed.

**Impact of SVD-based Graph Contrastive Learning**: Comparing ALEX with ALEX w/o G helps us understand the effectiveness of the SVD-based graph contrastive learning component. The results show that ALEX outperforms ALEX w/o G, which indicates that SVD-based graph contrastive learning benefits the model by extracting robust node representation bypassing the label noise.

**Impact of Balanced Domain Alignment**: To verify the effectiveness of the balanced domain alignment module, we compare ALEX and ALEX w/o B. From the results, we can observe that ALEX

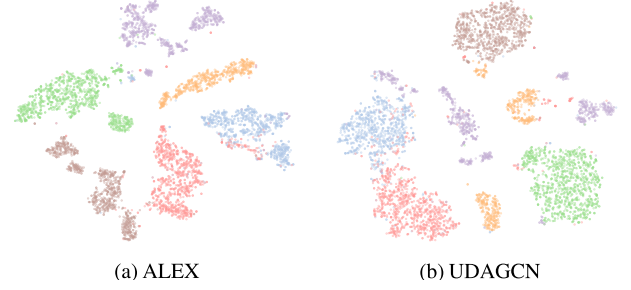

(a) ALEX         (b) UDAGCN

**Figure 5: Visualization of vector embedding with Dimensionality Reduction by t-SNE. Each point represents a node and the colors stand for different labels.**

performs significantly better than ALEX w/o B, indicating that the subgraph sampling with balanced domain alignment effectively addresses the problem of domain shift along with label shifts.

**Impact of Mutual Information-aware Refinement**: We compare ALEX with ALEX w/o M to investigate the effectiveness of the mutual information-aware refinement procedure, which identifies noisy nodes with high inconsistency. The results show that ALEX achieves better performance than ALEX w/o M in most datasets, demonstrating the importance of the mutual information-aware refinement procedure in improving the model's resilience to noise.

## 4.6 Visualization (RQ5)

Furthermore, we investigate the representational ability by visualizing the learned embedding, projected with T-distributed Stochastic Neighbor Embedding ($t$-SNE), in a 2-D space. Fig 5 compares the visualization results between our ALEX and best baseline UDAGCN for the 1A→D task with Pair Noise and default label noise rate. We can observe that Embeddings generated by our ALEX exhibit more clear and distinct boundary lines between different classes compared to the best baseline. This demonstrates that our ALEX can produce more meaningful graph embedding than other approaches, which explains our better classification performance.

## 5 CONCLUSION

This paper studies the problem of graph transfer learning under label noise, which aims to transfer knowledge from a noisy source graph to an unlabeled target graph. We propose a novel method named ALEX to solve the problem, which first generates robust node representations using graph contrastive learning. Then, we calculate the prior distribution for subgraph sampling for balanced domain alignment. Finally, we leverage the mutual information maximum to identify noisy samples which overfit noisy labels. Extensive experiments on various datasets substantiate the superiority of the proposed ALEX. In future work, we would extend our ALEX model to address more realistic challenges, including label imbalance, and zero-shot learning on graphs.

# REFERENCES

[1] Kirk Baker. 2005. Singular value decomposition tutorial. *The Ohio State University* 24 (2005).

[2] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. 2018. Mutual information neural estimation. In *Proceedings of the International Conference on Machine Learning*. 531–540.

[3] Xuheng Cai, Chao Huang, Lianghao Xia, and Xubin Ren. 2023. LightGCL: Simple Yet Effective Graph Contrastive Learning for Recommendation. *arXiv preprint arXiv:2302.08191* (2023).

[4] Ji Chang, Jing Li, Yu Kang, Wenjun Lv, Ting Xu, Zerui Li, Wei Xing Zheng, Hongwei Han, and Haining Liu. 2021. Unsupervised domain adaptation using maximum mean discrepancy optimization for lithology identification. *Geophysics* 86, 2 (2021), ID19–ID30.

[5] Yongqiang Chen, Yonggang Zhang, Yatao Bian, Han Yang, MA Kaili, Binghui Xie, Tongliang Liu, Bo Han, and James Cheng. 2022. Learning causally invariant representations for out-of-distribution generalization on graphs. (2022), 22131–22148.

[6] Enyan Dai, Charu Aggarwal, and Suhang Wang. 2021. NRGNN: Learning a label noise resistant graph neural network on sparsely and noisily labeled graphs. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 227–236.

[7] Quanyu Dai, Xiao-Ming Wu, Jiaren Xiao, Xiao Shen, and Dan Wang. 2022. Graph Transfer Learning via Adversarial Domain Adaptation with Graph Convolution. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[8] Xuefeng Du, Tian Bian, Yu Rong, Bo Han, Tongliang Liu, Tingyang Xu, Wenbing Huang, and Junzhou Huang. 2021. PI-GNN: A Novel Perspective on Semi-Supervised Node Classification against Noisy Labels. *arXiv preprint arXiv:2106.07451* (2021).

[9] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph neural networks for social recommendation. In *The world wide web conference*. 417–426.

[10] Wenqi Fan, Yao Ma, Qing Li, Jianping Wang, Guoyong Cai, Jiliang Tang, and Dawei Yin. 2020. A graph neural network framework for social recommendations. *IEEE Transactions on Knowledge and Data Engineering* 34, 5 (2020), 2033–2047.

[11] 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.

[12] Yunchao Gong, Qifa Ke, Michael Isard, and Svetlana Lazebnik. 2014. A multi-view embedding space for modeling internet images, tags, and their semantics. *International Journal of Computer Vision* 106 (2014), 210–233.

[13] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. 2011. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.* 53, 2 (2011), 217–288.

[14] William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 1025–1035.

[15] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Proceedings of the Conference on Neural Information Processing Systems*.

[16] Lei Ji, Rongcheng Tu, Kevin Lin, Lijuan Wang, and Nan Duan. 2022. Multimodal graph neural network for video procedural captioning. *Neurocomputing* 488 (2022), 88–96.

[17] Xiang Jiang, Qicheng Lao, Stan Matwin, and Mohammad Havaei. 2020. Implicit class-conditioned domain alignment for unsupervised domain adaptation. In *Proceedings of the International Conference on Machine Learning*. 4816–4827.

[18] Wei Ju, Zheng Fang, Yiyang Gu, Zequn Liu, Qingqing Long, Ziyue Qiao, Yifang Qin, Jianhao Shen, Fang Sun, Zhiping Xiao, et al. 2023. A Comprehensive Survey on Deep Graph Representation Learning. *arXiv preprint arXiv:2304.05055* (2023).

[19] Wei Ju, Xiao Luo, Meng Qu, Yifan Wang, Chong Chen, Minghua Deng, Xian-Sheng Hua, and Ming Zhang. 2023. TGNN: A joint semi-supervised framework for graph-level classification. *arXiv preprint arXiv:2304.11688* (2023).

[20] Wei Ju, Yifang Qin, Ziyue Qiao, Xiao Luo, Yifan Wang, Yanjie Fu, and Ming Zhang. 2022. Kernel-based substructure exploration for next POI recommendation. In *2022 IEEE International Conference on Data Mining (ICDM)*. IEEE, 221–230.

[21] Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR*.

[22] Jisoo Lee and Sae-Young Chung. 2020. Robust training with ensemble consensus. In *Proceedings of the International Conference on Learning Representations*.

[23] Jinfeng Li, Weifeng Liu, Yicong Zhou, Jun Yu, Dapeng Tao, and Changsheng Xu. 2022. Domain-invariant graph for adaptive semi-supervised domain adaptation. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 18, 3 (2022), 1–18.

[24] Mengxue Li, Yi-Ming Zhai, You-Wei Luo, Peng-Fei Ge, and Chuan-Xian Ren. 2020. Enhanced transport distance for unsupervised domain adaptation. In *Proceedings*

[25] of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 13936–13944.

[25] Shuang Li, Chi Harold Liu, Binhui Xie, Limin Su, Zhengming Ding, and Gao Huang. 2019. Joint adversarial domain adaptation. In *Proceedings of the ACM International Conference on Multimedia*. 729–737.

[26] Xiaofeng Liu, Zhenhua Guo, Site Li, Fangxu Xing, Jane You, C-C Jay Kuo, Georges El Fakhri, and Jonghye Woo. 2021. Adversarial unsupervised domain adaptation with conditional and label shift: Infer, align and iterate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 10367–10376.

[27] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. 2018. Conditional adversarial domain adaptation. *Advances in neural information processing systems* 31 (2018).

[28] Xiao Luo, Wei Ju, Meng Qu, Yiyang Gu, Chong Chen, Minghua Deng, Xian-Sheng Hua, and Ming Zhang. 2022. Clear: Cluster-enhanced contrast for self-supervised graph representation learning. *IEEE Transactions on Neural Networks and Learning Systems* (2022).

[29] Xiao Luo, Daqing Wu, Zeyu Ma, Chong Chen, Minghua Deng, Jinwen Ma, Zhongming Jin, Jianqiang Huang, and Xian-Sheng Hua. 2020. Cimon: Towards high-quality hash codes. *arXiv preprint arXiv:2010.07804* (2020).

[30] Xiao Luo, Jingyang Yuan, Zijie Huang, Huiyu Jiang, Yifang Qin, Wei Ju, Ming Zhang, and Yizhou Sun. 2023. HOPE: High-order Graph ODE For Modeling Interacting Dynamics. (2023).

[31] Yadan Luo, Zi Huang, Zijian Wang, Zheng Zhang, and Mahsa Baktashmotlagh. 2020. Adversarial bipartite graph learning for video domain adaptation. In *Proceedings of the ACM International Conference on Multimedia*. 19–27.

[32] Hoang NT, Choong Jun Jin, and Tsuyoshi Murata. 2019. Learning graph neural networks with noisy labels. *arXiv preprint arXiv:1905.01591* (2019).

[33] Youngmin Oh, Beomjun Kim, and Bumsub Ham. 2021. Background-Aware Pooling and Noise-Aware Loss for Weakly-Supervised Semantic Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[34] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. 2017. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[35] Shichao Pei, Lu Yu, Guoxian Yu, and Xiangliang Zhang. 2020. Rea: Robust cross-lingual entity alignment between knowledge graphs. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2175–2184.

[36] 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.

[37] Siyi Qian, Haochao Ying, Renjun Hu, Jingbo Zhou, Jintai Chen, Danny Z Chen, and Jian Wu. 2023. Robust training of graph neural networks via noise governance. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*. 607–615.

[38] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. 2015. Training deep neural networks on noisy labels with bootstrapping. In *Proceedings of the International Conference on Learning Representations*.

[39] Amirreza Salamat, Xiao Luo, and Ali Jafari. 2021. HeteroGraphRec: A heterogeneous graph-based neural networks for social recommendations. *Knowledge-Based Systems* 217 (2021), 106817.

[40] 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.

[41] Cheng Tan, Jun Xia, Lirong Wu, and Stan Z Li. 2021. Co-learning: Learning from noisy labels with self-supervision. In *Proceedings of the ACM International Conference on Multimedia*. 1405–1413.

[42] 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.

[43] Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 2008. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 990–998.

[44] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. 2017. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7167–7176.

[45] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. In *International Conference on Learning Representations*.

[46] Chenyu Wang, Zongyu Lin, Xiaochen Yang, Jiao Sun, Mingxuan Yue, and Cyrus Shahabi. 2022. Hagen: Homophily-aware graph convolutional recurrent network for crime forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 4193–4200.

[47] Qifan Wang, Yinwei Wei, Jianhua Yin, Jianlong Wu, Xuemeng Song, and Liqiang Nie. 2021. Dualgnn: Dual graph neural network for multimedia recommendation. *IEEE Transactions on Multimedia* (2021).

[48] Tiantian Wang, Sifei Liu, Yapeng Tian, Kai Li, and Ming-Hsuan Yang. 2021. Video matting via consistency-regularized graph neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 4902–4911.

[49] Wenguan Wang, Xiankai Lu, Jianbing Shen, David J Crandall, and Ling Shao. 2019. Zero-shot video object segmentation via attentive graph neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9236–9245.

[50] Xiao Wang, Hongrui Liu, Chuan Shi, and Cheng Yang. 2021. Be confident! towards trustworthy graph neural networks via confidence calibration. In *Proceedings of the Conference on Neural Information Processing Systems*. 23768–23779.

[51] Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *ICML*.

[52] 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.

[53] Man Wu, Shirui Pan, and Xingquan Zhu. 2022. Attraction and Repulsion: Unsupervised Domain Adaptive Graph Contrastive Learning Network. *IEEE Transactions on Emerging Topics in Computational Intelligence* (2022).

[54] Ying-Xin Wu, Xiang Wang, An Zhang, Xiangnan He, and Tat-Seng Chua. 2022. Discovering invariant rationales for graph neural networks. In *Proceedings of the International Conference on Machine Learning*.

[55] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems* (2020).

[56] Louis-Pascal Xhonneux, Meng Qu, and Jian Tang. 2020. Continuous graph neural networks. In *International Conference on Machine Learning*. PMLR, 10432–10441.

[57] Zizheng Yan, Yushuang Wu, Guanbin Li, Yipeng Qin, Xiaoguang Han, and Shuguang Cui. 2022. Multi-level Consistency Learning for Semi-supervised Domain Adaptation. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, Lud De Raedt (Ed.). International Joint Conferences on Artificial Intelligence Organization, 1530–1536. https://doi.org/10.24963/ijcai.2022/213 Main Track.

[58] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. 2019. How does disagreement help generalization against label corruption?. In *International Conference on Machine Learning*. PMLR, 7164–7173.

[59] Hao Yuan, Jiliang Tang, Xia Hu, and Shuiwang Ji. 2020. Xgnn: Towards model-level explanations of graph neural networks. In *Proceedings of the International ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 430–438.

[60] Jingyang Yuan, Xiao Luo, Yifang Qin, Yusheng Zhao, Wei Ju, and Ming Zhang. 2023. Learning on Graphs under Label Noise. In *IEEE International Conference on Acoustics, Speech and Signal Processing*. 1–5.

[61] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2021. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM* (2021).

[62] HaiYang Zhang, XiMing Xing, and Liang Liu. 2021. Dualgraph: A graph-based method for reasoning about label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9654–9663.

[63] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. 2018. Collaborative and adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3801–3809.

[64] Wenyu Zhang, Li Shen, Wanyue Zhang, and Chuan-Sheng Foo. 2022. Few-Shot Adaptation of Pre-Trained Networks for Domain Shift. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, Lud De Raedt (Ed.). International Joint Conferences on Artificial Intelligence Organization, 1665–1671. https://doi.org/10.24963/ijcai.2022/232 Main Track.

[65] Xiaowen Zhang, Yuntao Du, Rongbiao Xie, and Chongjun Wang. 2021. Adversarial Separation Network for Cross-Network Node Classification. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 2618–2626.

[66] Yanfu Zhang, Shangqian Gao, Jian Pei, and Heng Huang. 2022. Improving Social Network Embedding via New Second-Order Continuous Graph Neural Networks. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2515–2523.

[67] Qi Zhu, Carl Yang, Yidan Xu, Haonan Wang, Chao Zhang, and Jiawei Han. 2021. Transfer learning of graph neural networks with ego-graph information maximization. In *Proceedings of the Conference on Neural Information Processing Systems*. 1766–1779.