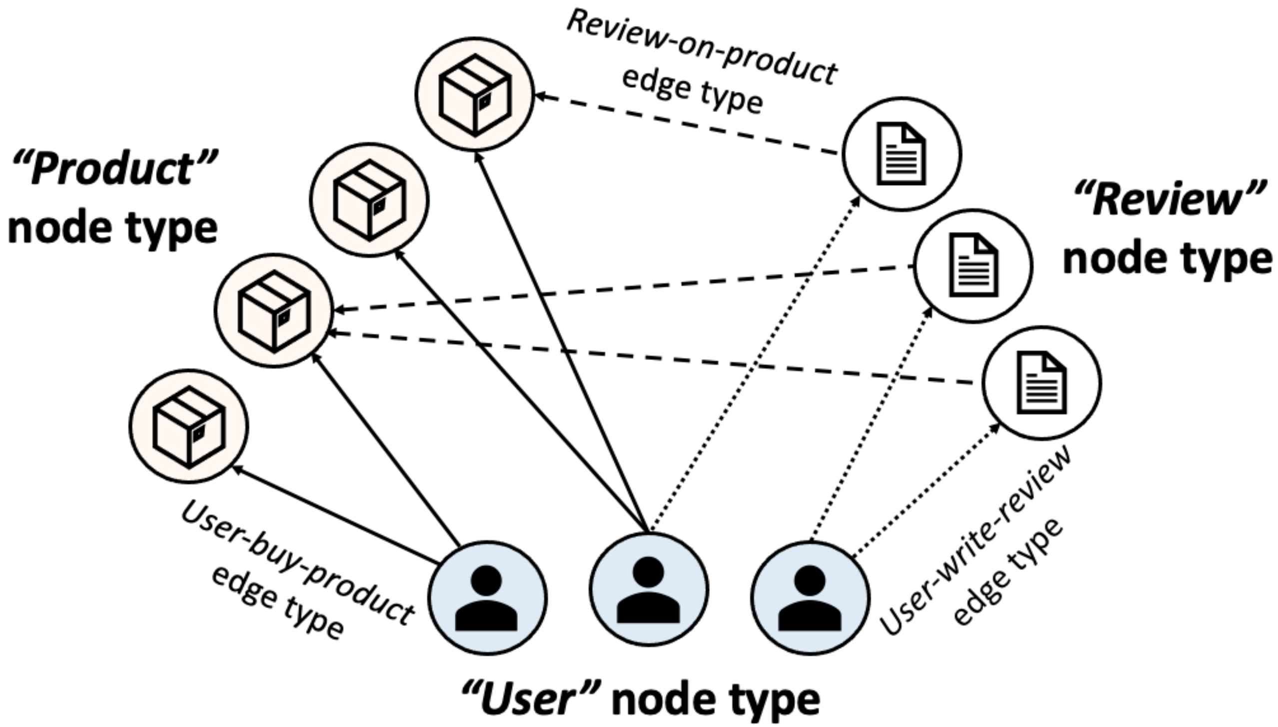
Zero-shot Transfer Learning within a Heterogeneous Graph via Knowledge Transfer Networks

Minji Yoon, John Palowitch, Dustin Zelle,
Ziniu Hu, Ruslan Salakhutdinov, Bryan Perozzi

Heterogeneous Graphs (HG)

- Composed of multiple types of nodes and edges
- EX) e-commerce networks



Heterogeneous Graph Neural Networks (HGNNs)

- MPNNs for homogeneous graphs:

$$h_j^{(l)} = \alpha(\underbrace{\boxed{W^{(l)}}}_{\text{Transformation parameters}} \cdot (\frac{1}{|\mathcal{E}(j)|} \sum_{e \in \mathcal{E}(j)} \underbrace{\boxed{M^{(l)}}}_{\text{Message parameters}} \cdot \left(h_i^{(l-1)} \parallel h_j^{(l-1)}\right))$$

Heterogeneous Graph Neural Networks (HGNNs)

- MPNNs for homogeneous graphs:

$$h_j^{(l)} = \alpha\left(\boxed{W^{(l)}} \cdot \left(\frac{1}{|\mathcal{E}(j)|} \sum_{e \in \mathcal{E}(j)} \boxed{M^{(l)}} \cdot \left(h_i^{(l-1)} \parallel h_j^{(l-1)}\right)\right)\right)$$

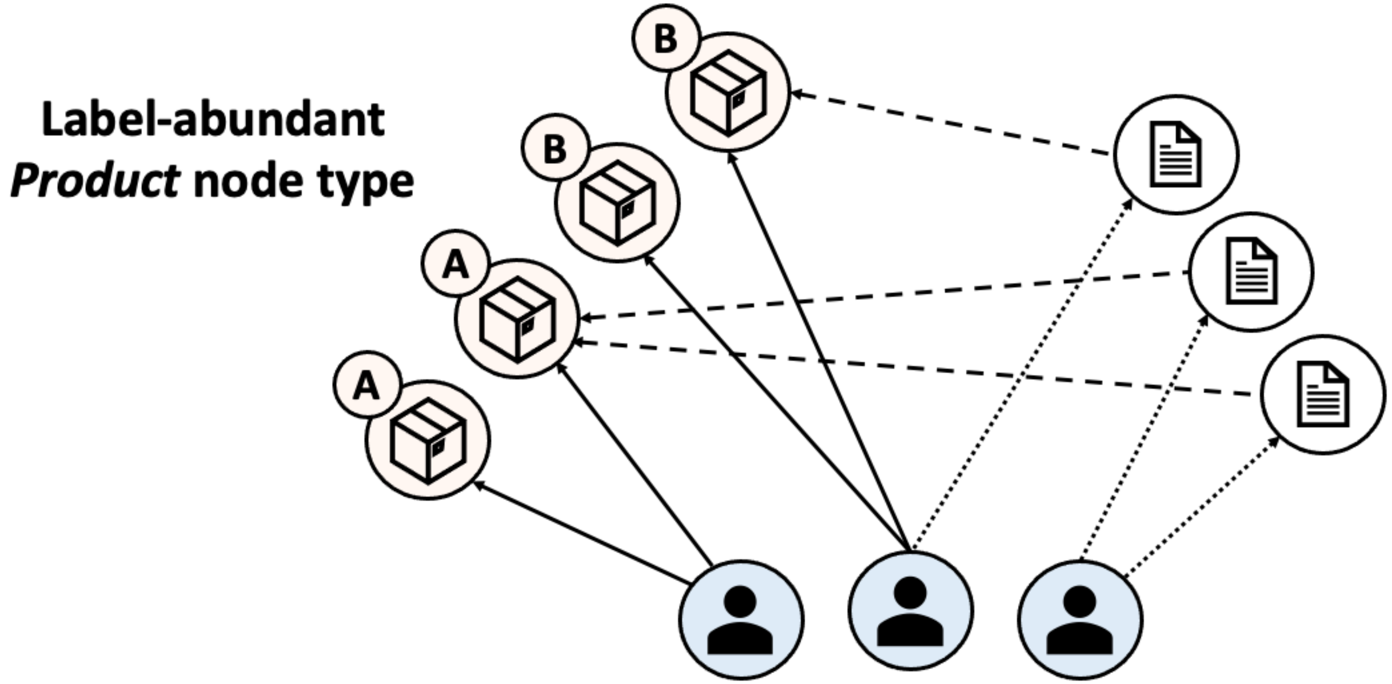
- H-MPNNs for heterogeneous graphs:

$$h_j^{(l)} = \alpha\left(\boxed{W_{\tau(j)}^{(l)}} \cdot \left(\big\Vert_{r \in \mathcal{R}} \frac{1}{|\mathcal{E}_r(j)|} \sum_{e \in \mathcal{E}_r(j)} \boxed{M_{\phi((i,j))}^{(l)}} \cdot \left(h_i^{(l-1)} \parallel h_j^{(l-1)}\right)\right)\right)$$

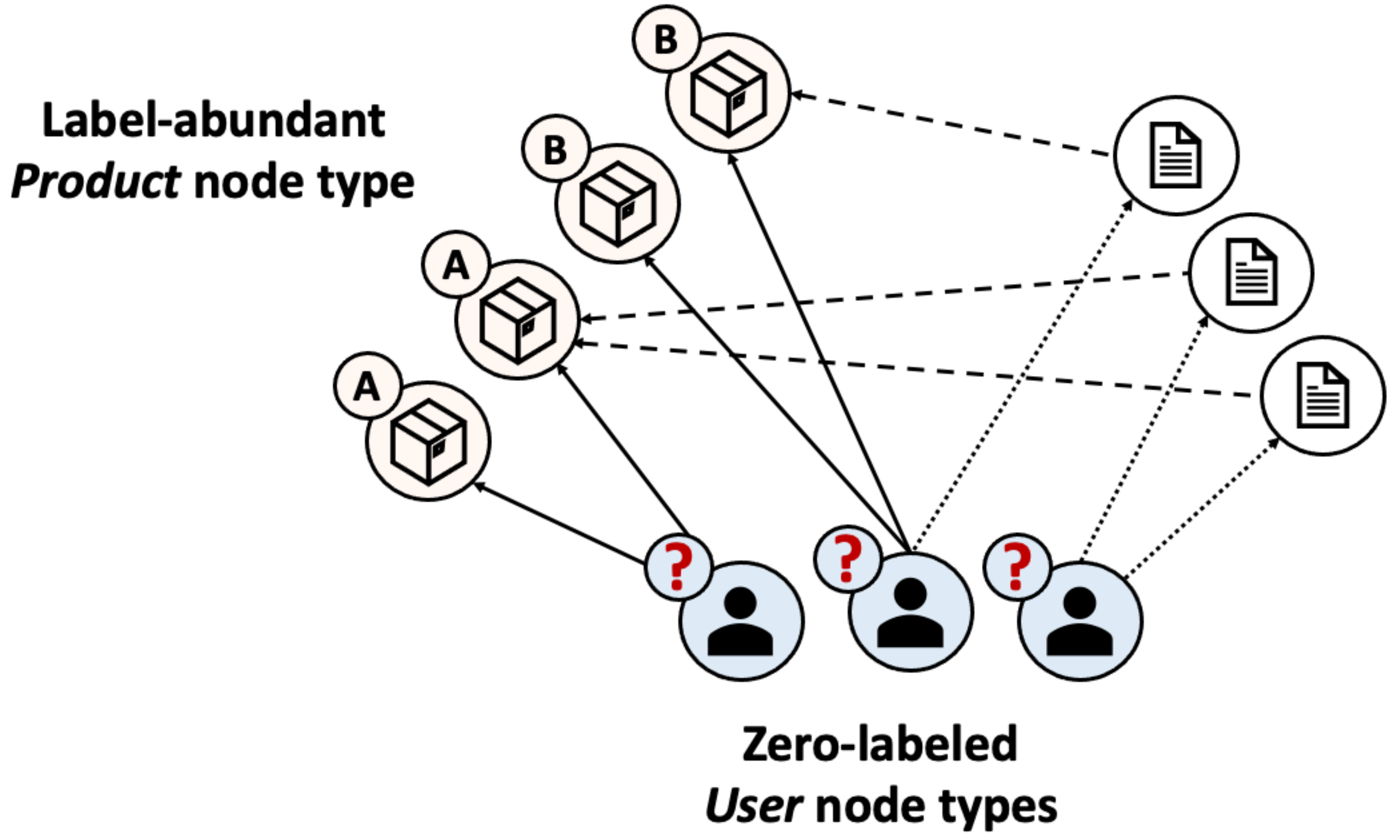
Node-type-specific transformation parameters

Edge-type-specific message parameters

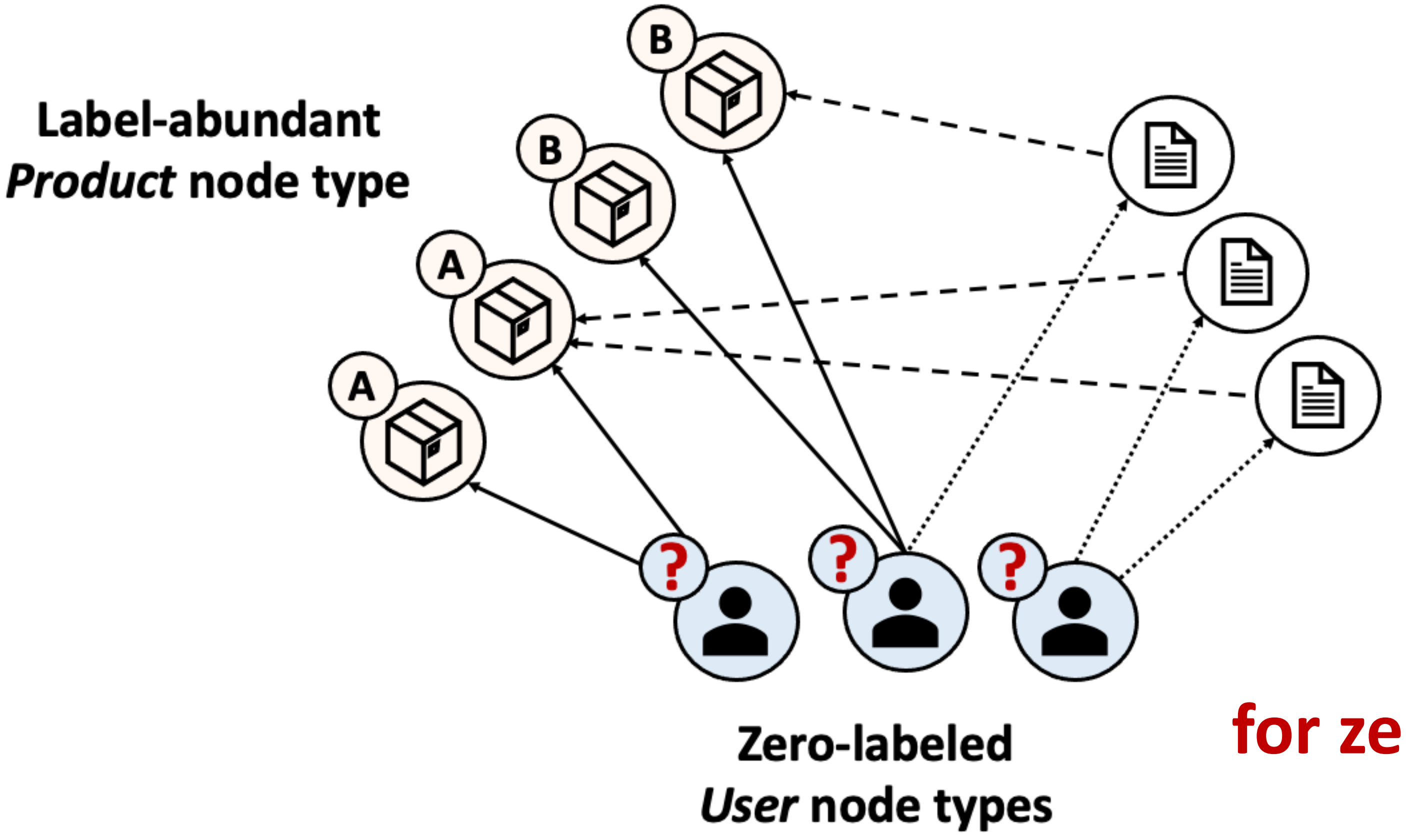
Label imbalance between node types



Label imbalance between node types

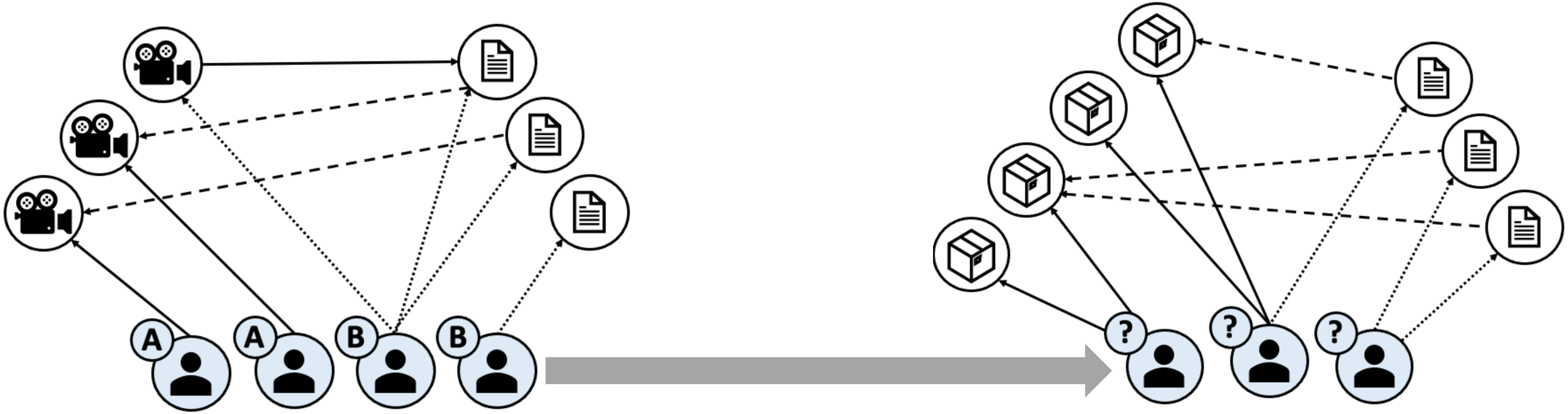


Label imbalance between node types



**How can we train
a HGNN model
for zero-labeled node types?**

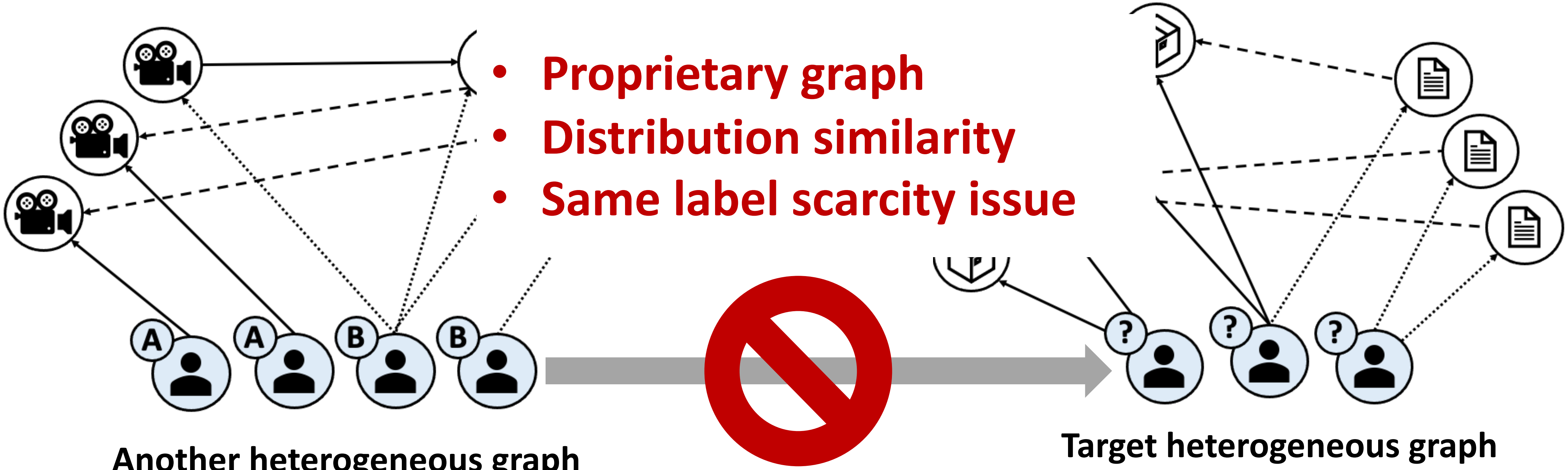
Previous approach:
Graph-to-Graph Transfer Learning



Another heterogeneous graph with abundant *User* labels

Target heterogeneous graph with no *User* labels

Previous approach:
Graph-to-Graph Transfer Learning

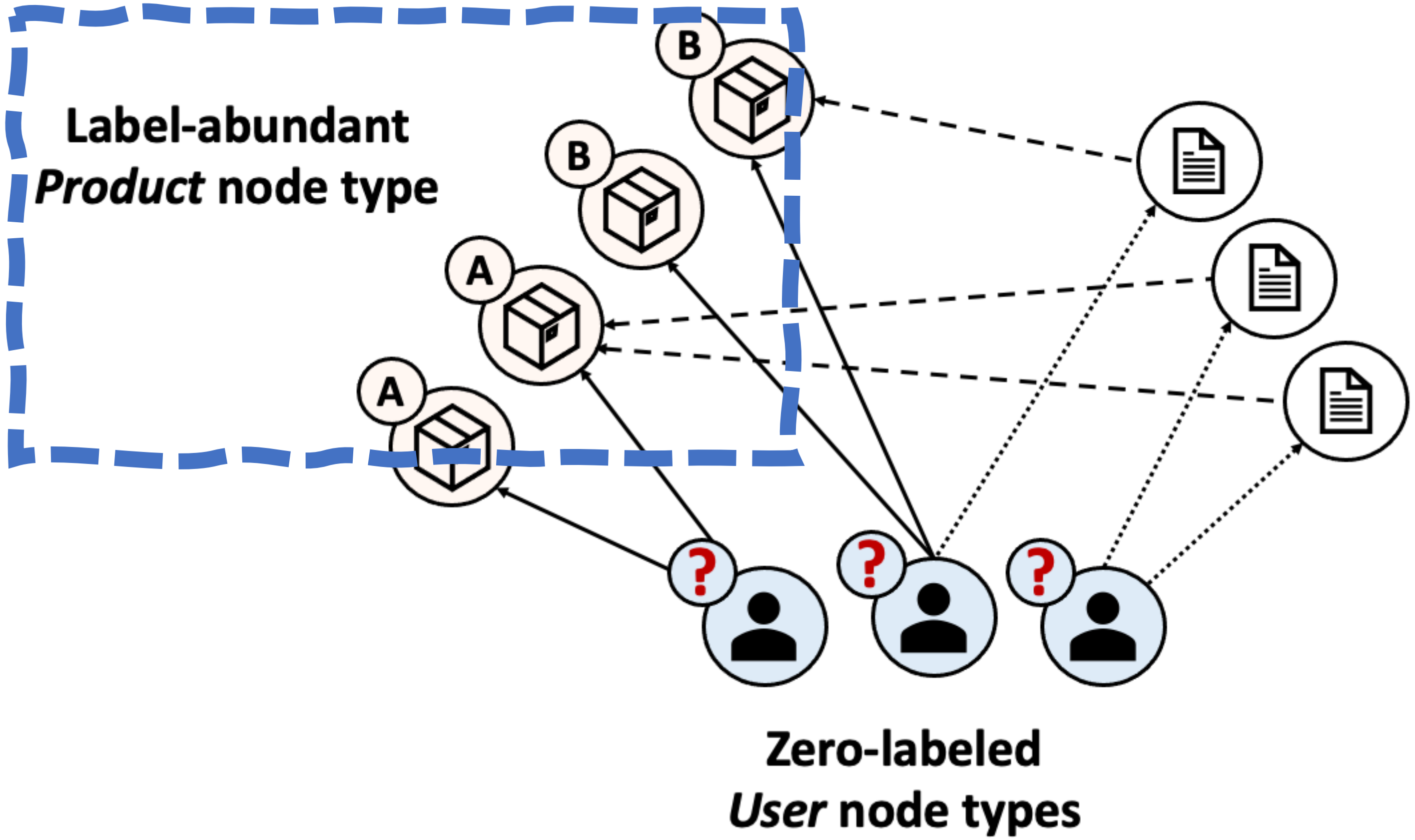


- Proprietary graph
- Distribution similarity
- Same label scarcity issue

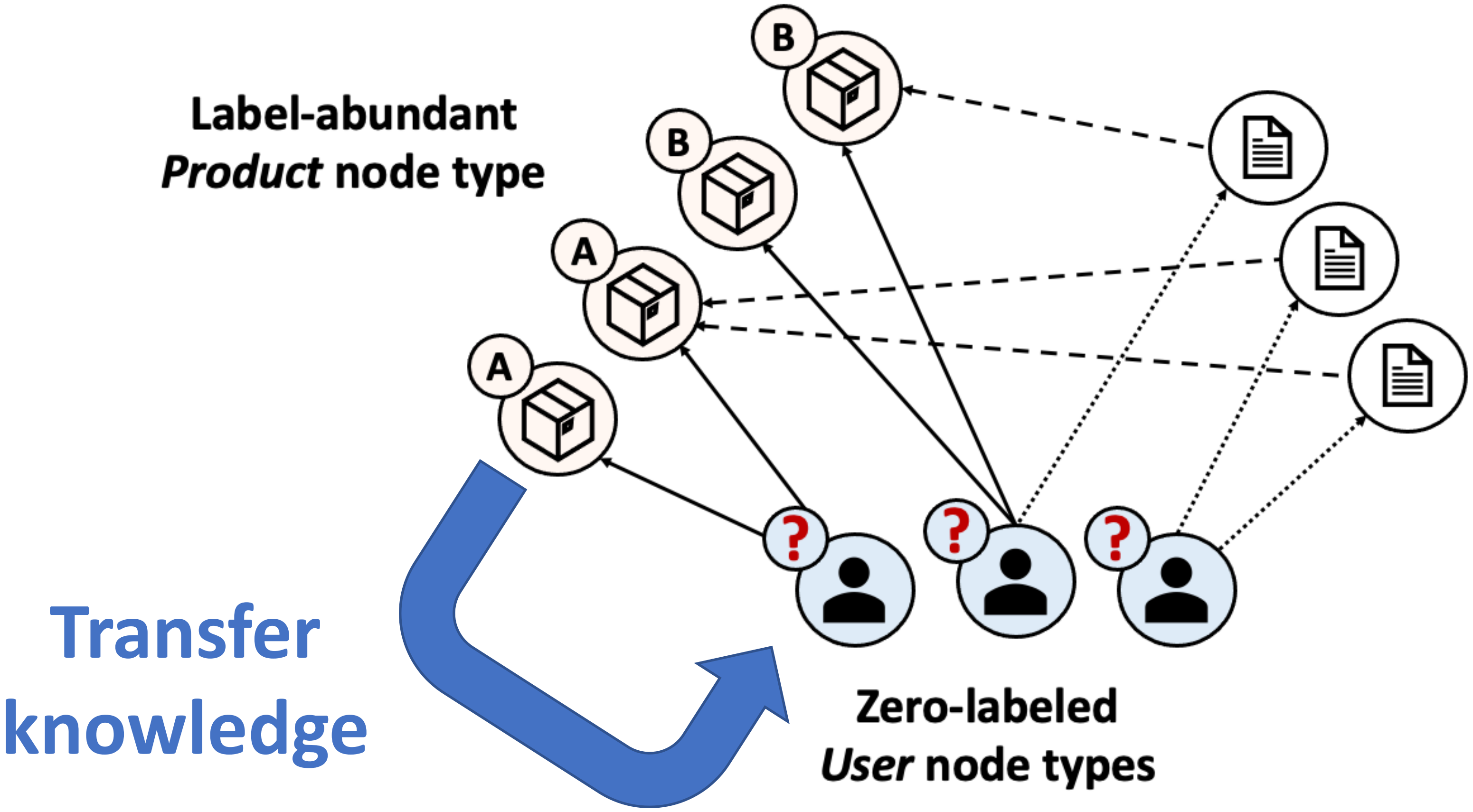
Another heterogeneous graph with abundant *User* labels

Target heterogeneous graph with no *User* labels

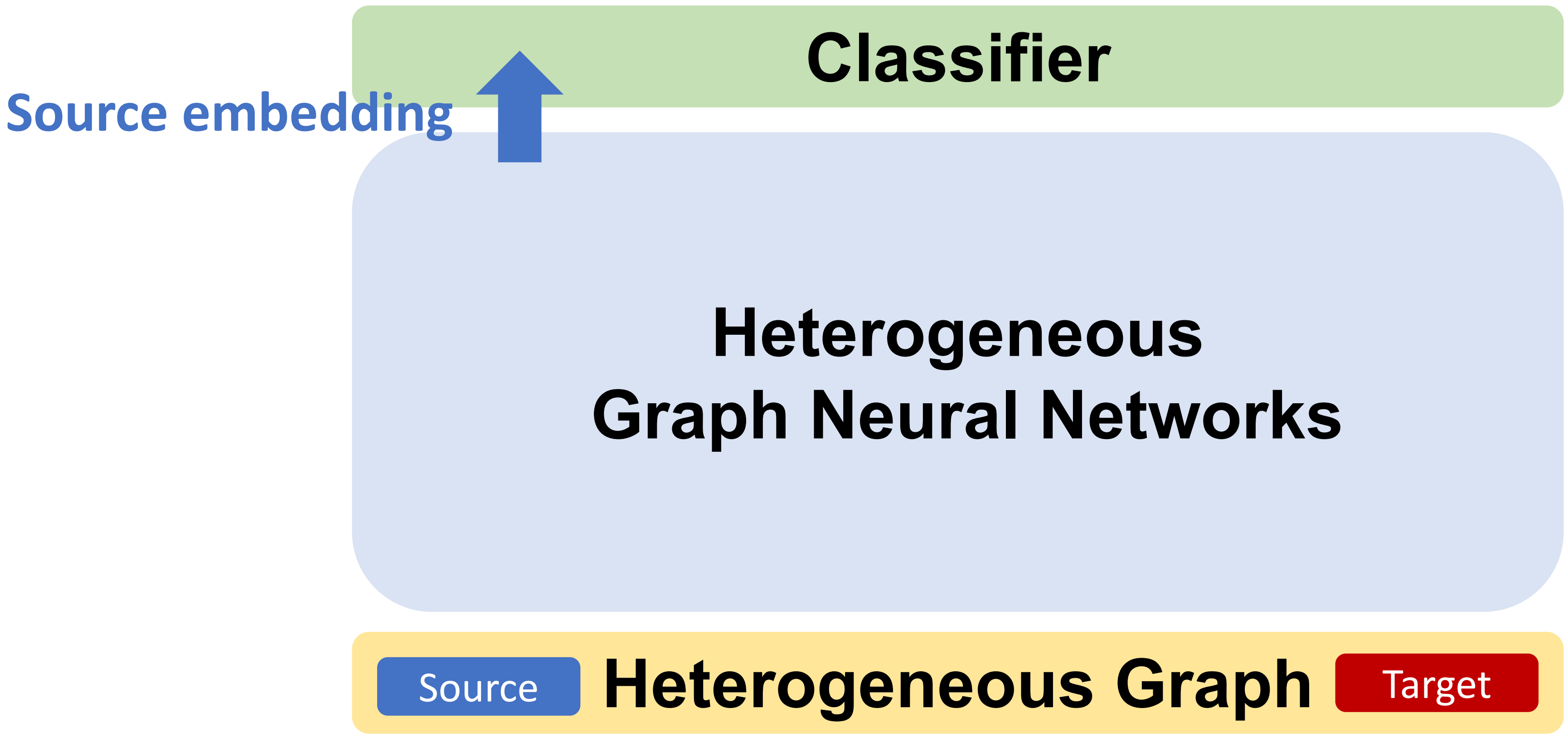
Transfer Learning within a Heterogeneous Graph



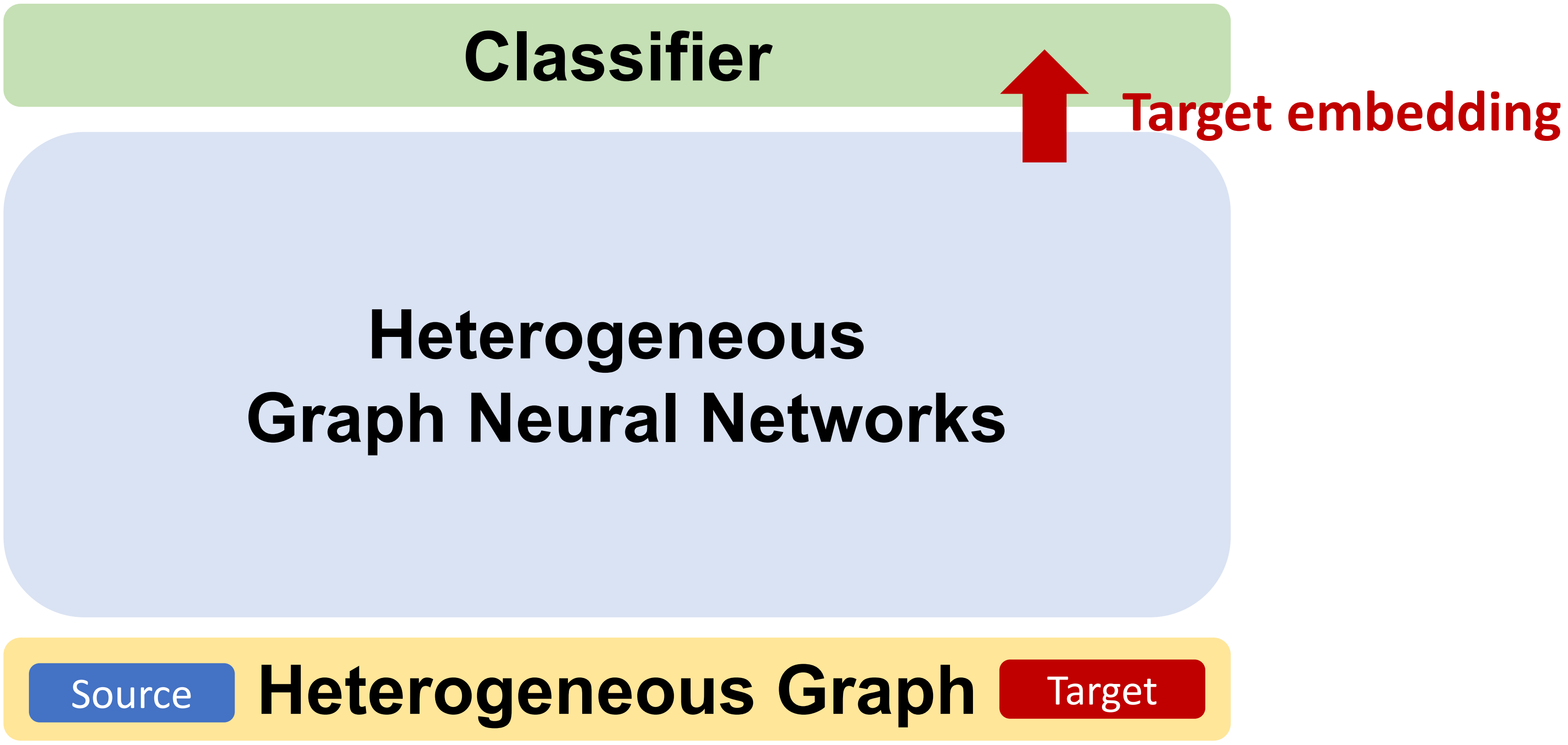
Transfer Learning within a Heterogeneous Graph



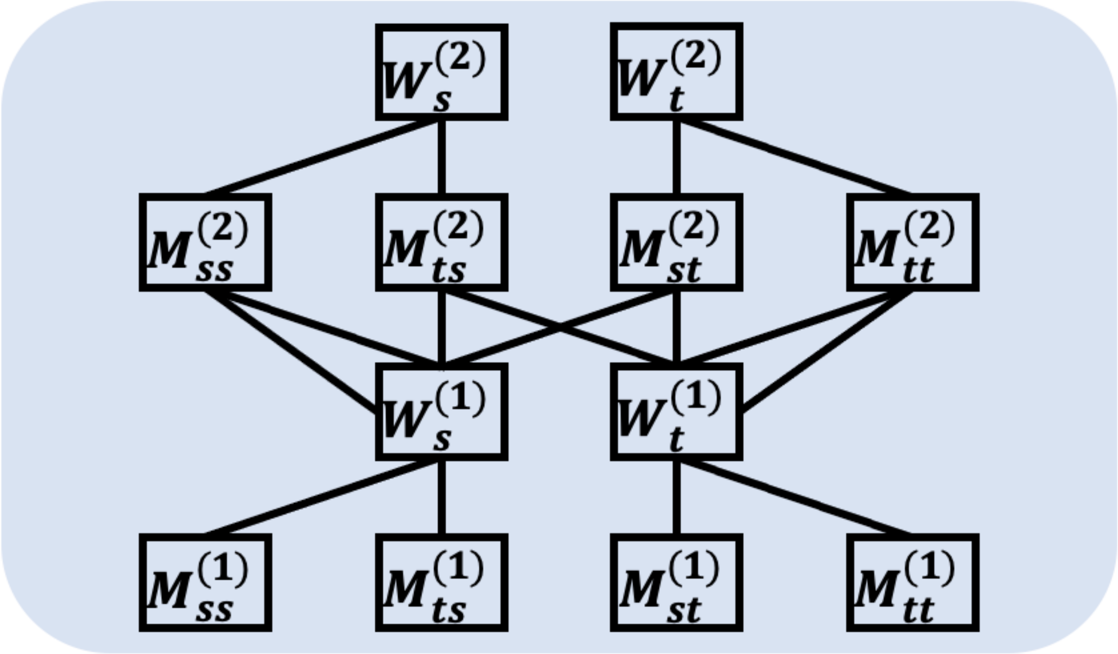
Is this problem challenging?



Is this problem challenging?

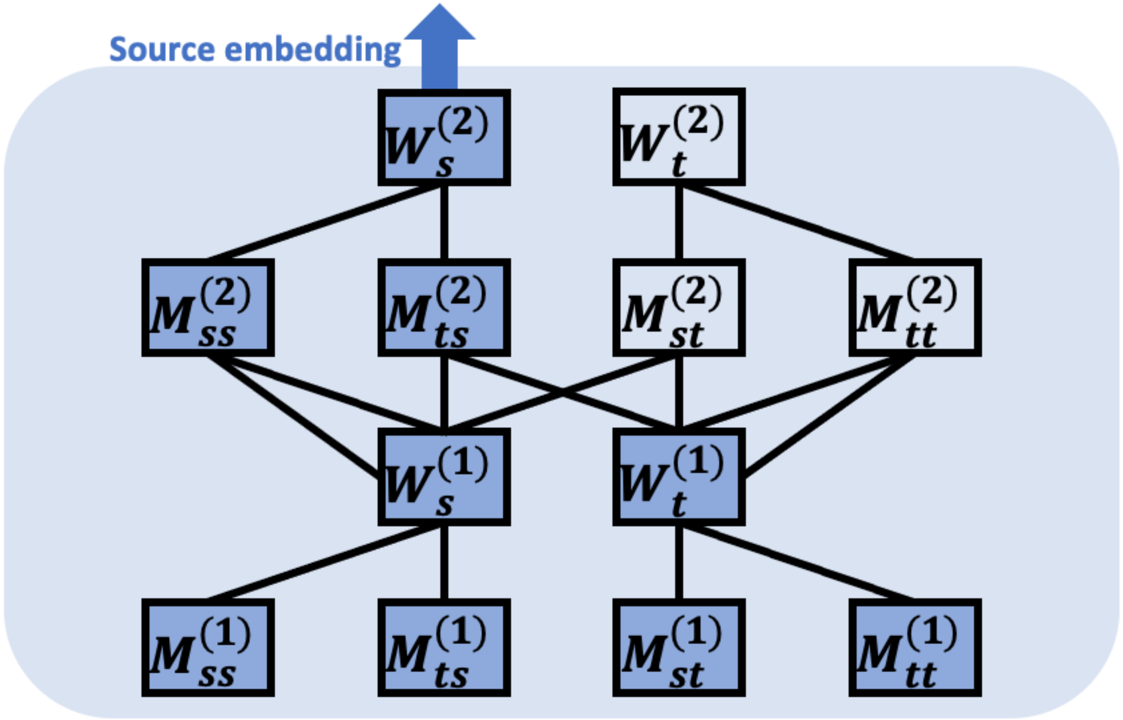


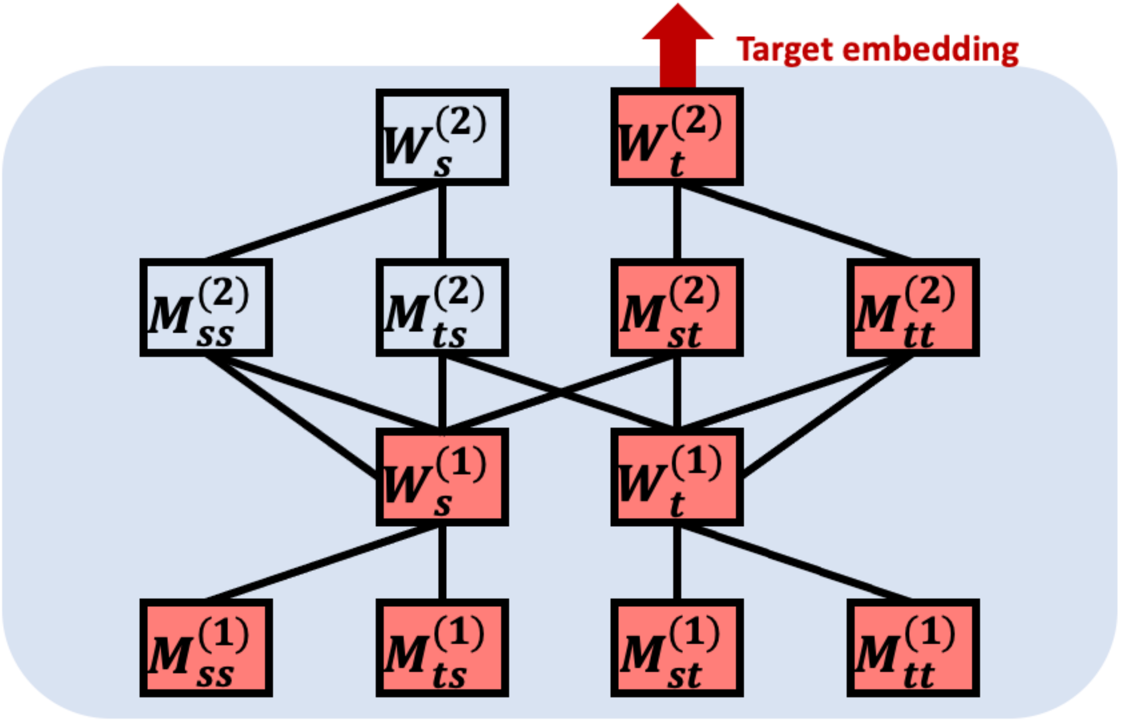
Distinct Feature Extractors in HGNNs



Yoon et al., Zero-shot Transfer Learning within a Heterogeneous Graph via Knowledge Transfer Networks, NeurIPS'22

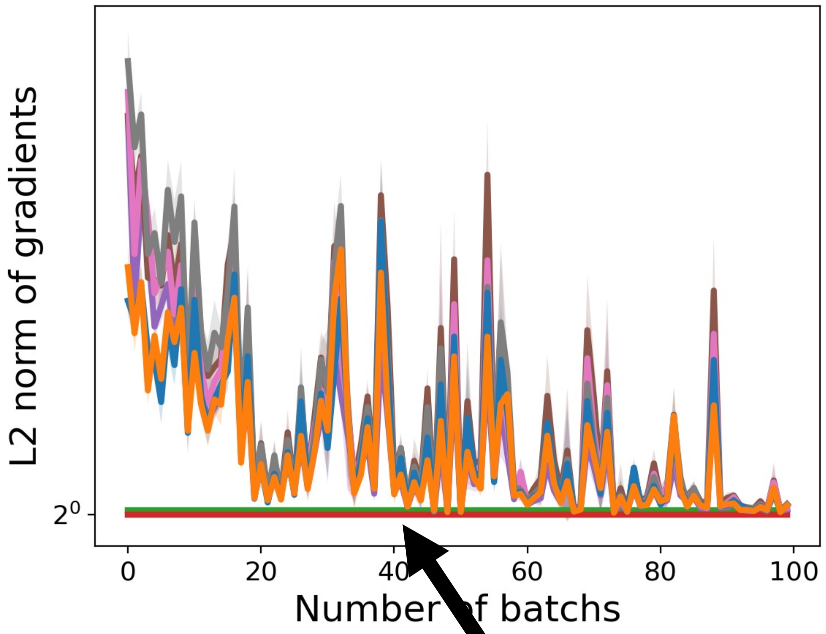
Distinct Feature Extractors in HGNNs

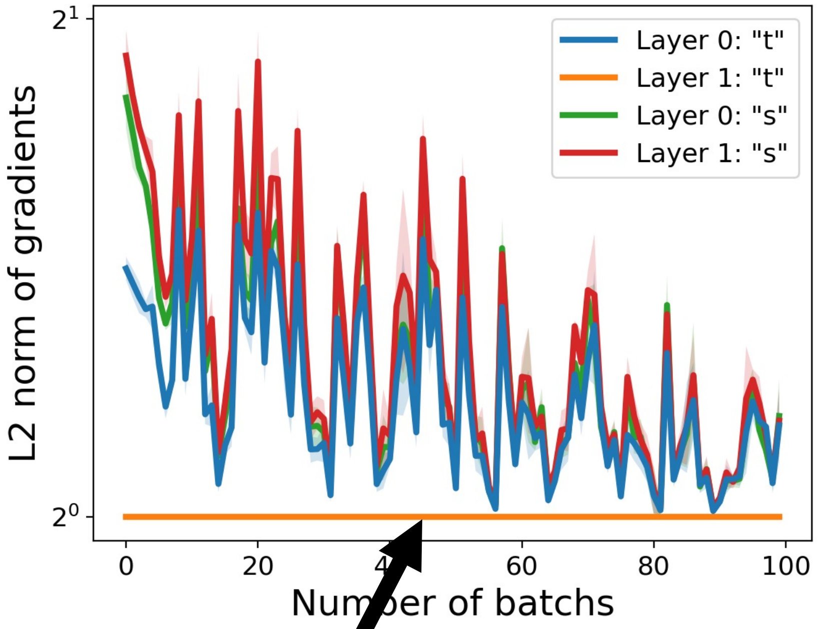




Yoon et al., Zero-shot Transfer Learning within a Heterogeneous Graph
via Knowledge Transfer Networks, NeurIPS'22

Distinct Feature Extractors in HGNNs



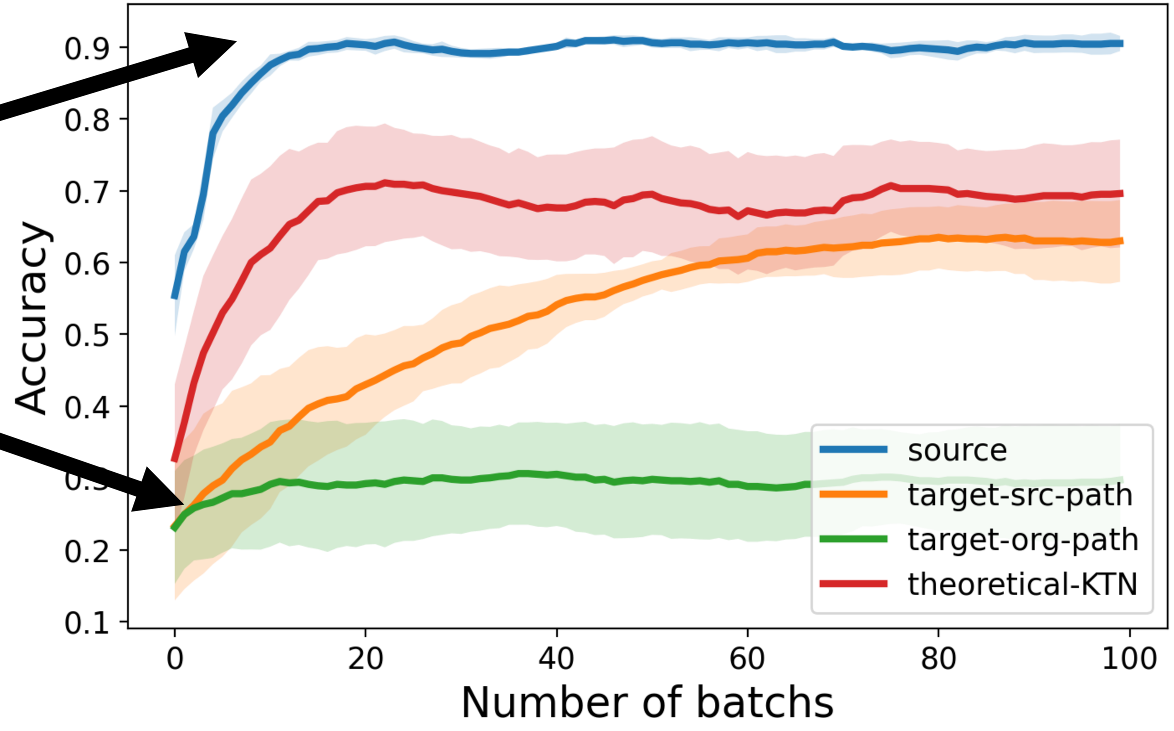


Small or no gradients for target-type parameters

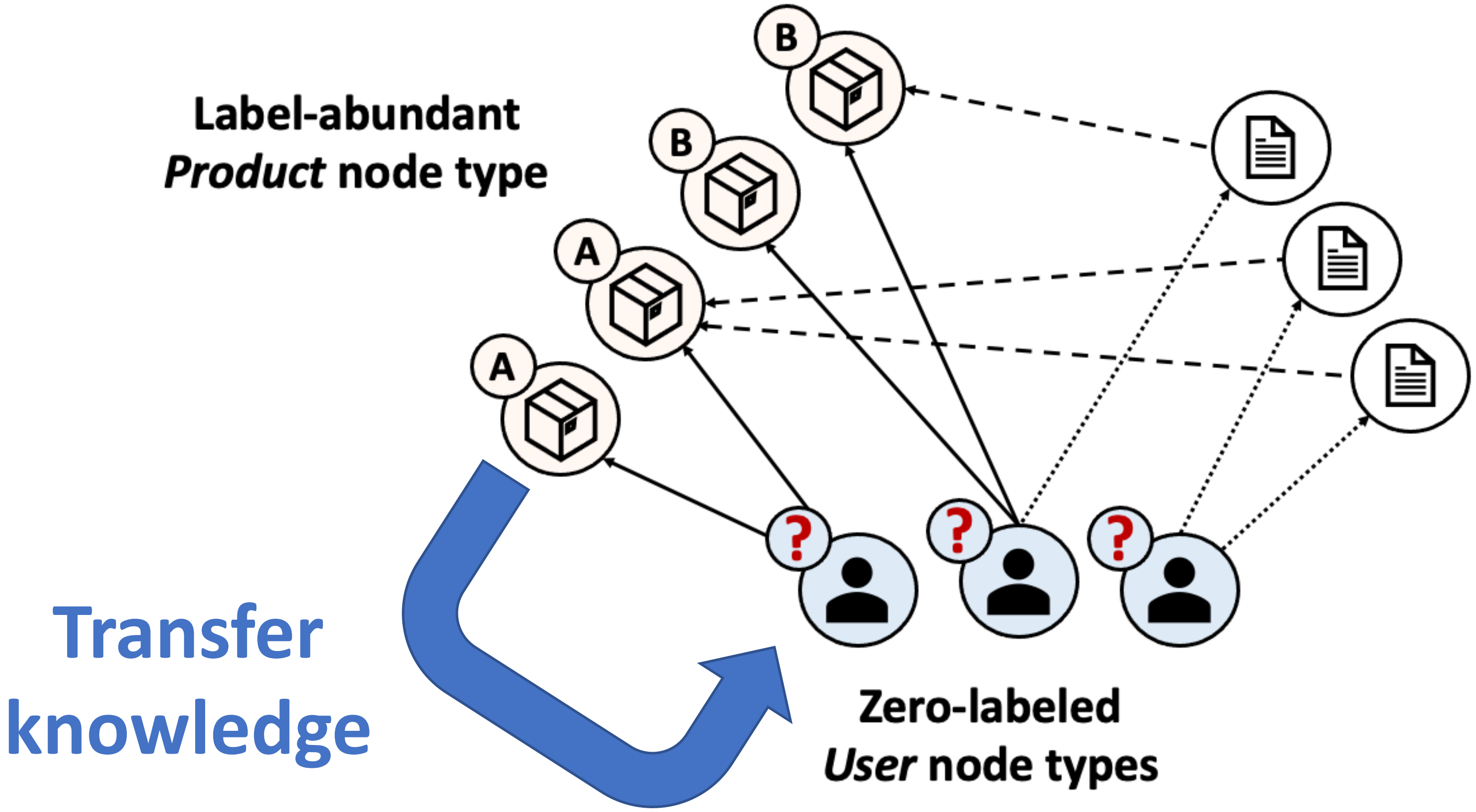
Yoon et al., Zero-shot Transfer Learning within a Heterogeneous Graph via Knowledge Transfer Networks, NeurIPS'22

Distinct Feature Extractors in HGNNs

Poor target performance

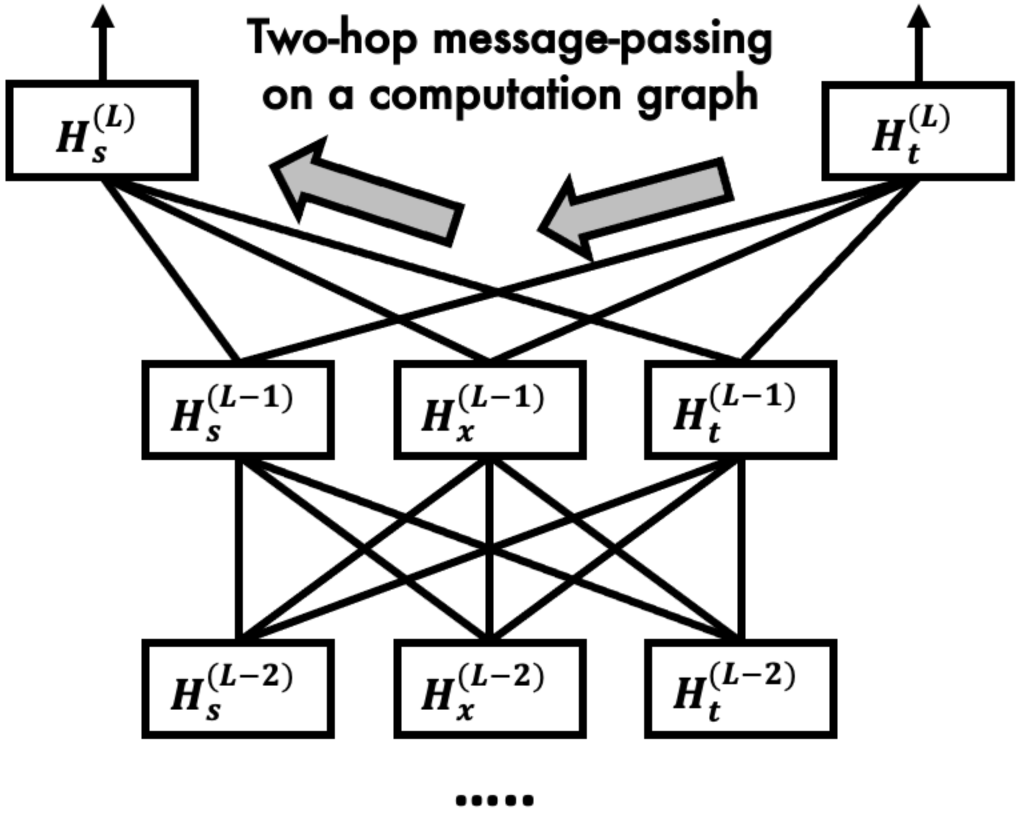


Then.. How can we solve this problem?



Yoon et al., Zero-shot Transfer Learning within a Heterogeneous Graph via Knowledge Transfer Networks, NeurIPS'22

Hints: Relationship between Feature Extractors



Yoon et al., Zero-shot Transfer Learning within a Heterogeneous Graph via Knowledge Transfer Networks, NeurIPS'22

Theoretically-induced
Mapping Function between Feature Extractors

Theorem 1. *Given a heterogeneous graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{T}, \mathcal{R}\}$. For any layer $l > 0$, define the set of* $(l-1)$-th *layer HMPNN parameters as*

$$\mathcal{Q}^{(l-1)} = \{M_r^{(l-1)} : r \in \mathcal{R}\} \cup \{W_t^{(l-1)} : t \in \mathcal{T}\}. \tag{9}$$

Let A be the total $n \times n$ adjacency matrix. Then for any $s, t \in \mathcal{T}$ there exist matrices $A^*_{ts} = a_{ts}(A)$ and $Q^*_{ts} = q_{ts}(\mathcal{Q}^{(l-1)})$ such that

$$H_s^{(l)} = A^*_{ts} H_t^{(l)} \boxed{Q^*_{ts}} \tag{10}$$

where $a_{ts}(\cdot)$ and $q_{ts}(\cdot)$ are matrix functions that depend only on s, t.

Hand-computed mapping functions

Proposed method:
Knowledge Transfer Networks (KTN)

$$\mathbf{t}_{\text{KTN}}(H_t^{(L)}) = A_{ts} H_t^{(L)} \boxed{T_{ts}}$$
$$\mathcal{L}_{\text{KTN}} = \left\| H_s^{(L)} - \mathbf{t}_{\text{KTN}}(H_t^{(L)}) \right\|_2$$

Learnable
mapping functions

Proposed method:
Knowledge Transfer Networks (KTN)

$$\mathbf{t}_{\text{KTN}}(H_t^{(L)}) = A_{ts} H_t^{(L)} T_{ts}$$

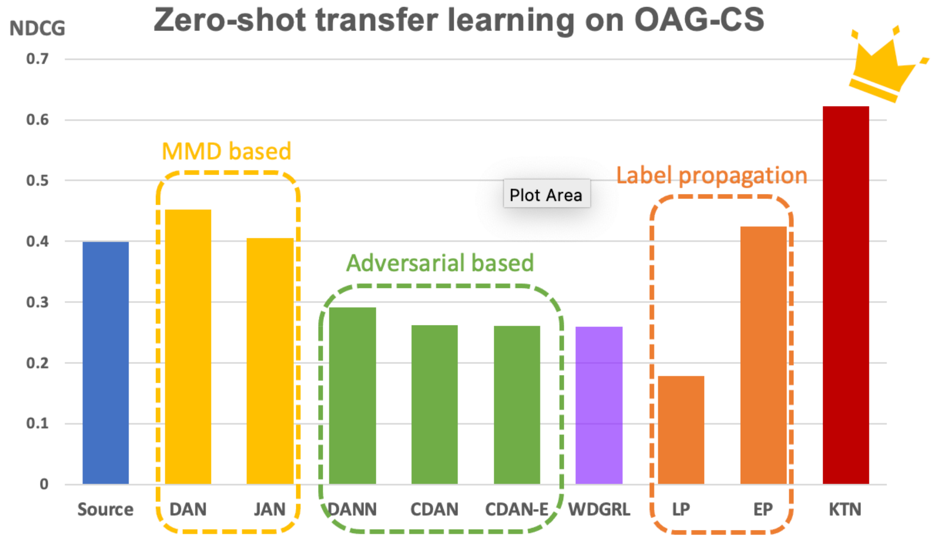
$$\mathcal{L}_{\text{KTN}} = \left\| H_s^{(L)} - \mathbf{t}_{\text{KTN}}(H_t^{(L)}) \right\|_2$$

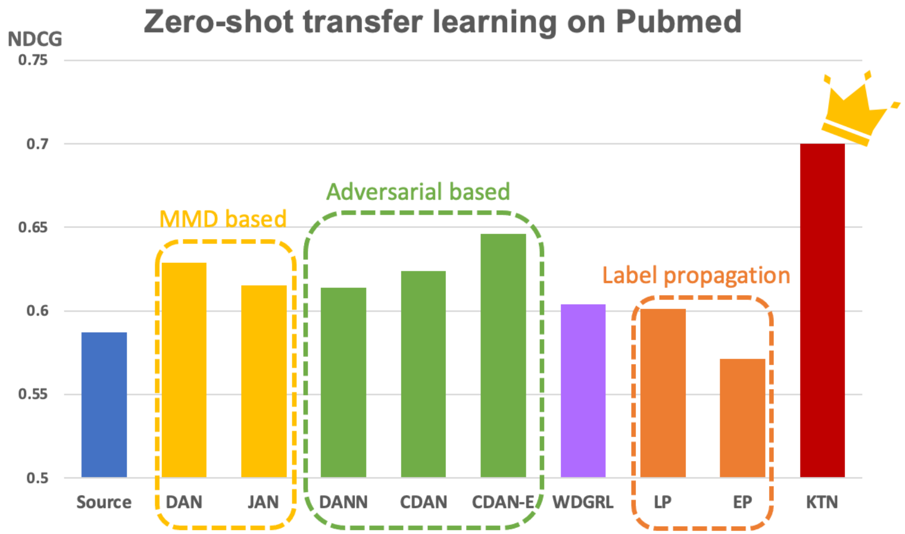
$$\min_{\mathbf{f},\, \mathbf{g},\, \mathbf{t}_{\text{KTN}}} \mathcal{L}_{\text{CL}}(\mathbf{g}(\mathbf{f}(\mathcal{G}, \mathcal{X})_s), \mathcal{Y}_s) + \boxed{\lambda \left\| \mathbf{f}(\mathcal{G}, \mathcal{X})_s - \mathbf{t}_{\text{KTN}}(\mathbf{f}(\mathcal{G}, \mathcal{X})_t) \right\|_2}$$

Knowledge transfer loss

Experiment (1) Zero-shot Transfer Learning

- 18 different tasks
- 6 SOTA zero-shot transfer learning baselines
- 2 traditional label propagation baselines
- 73% higher in MRR

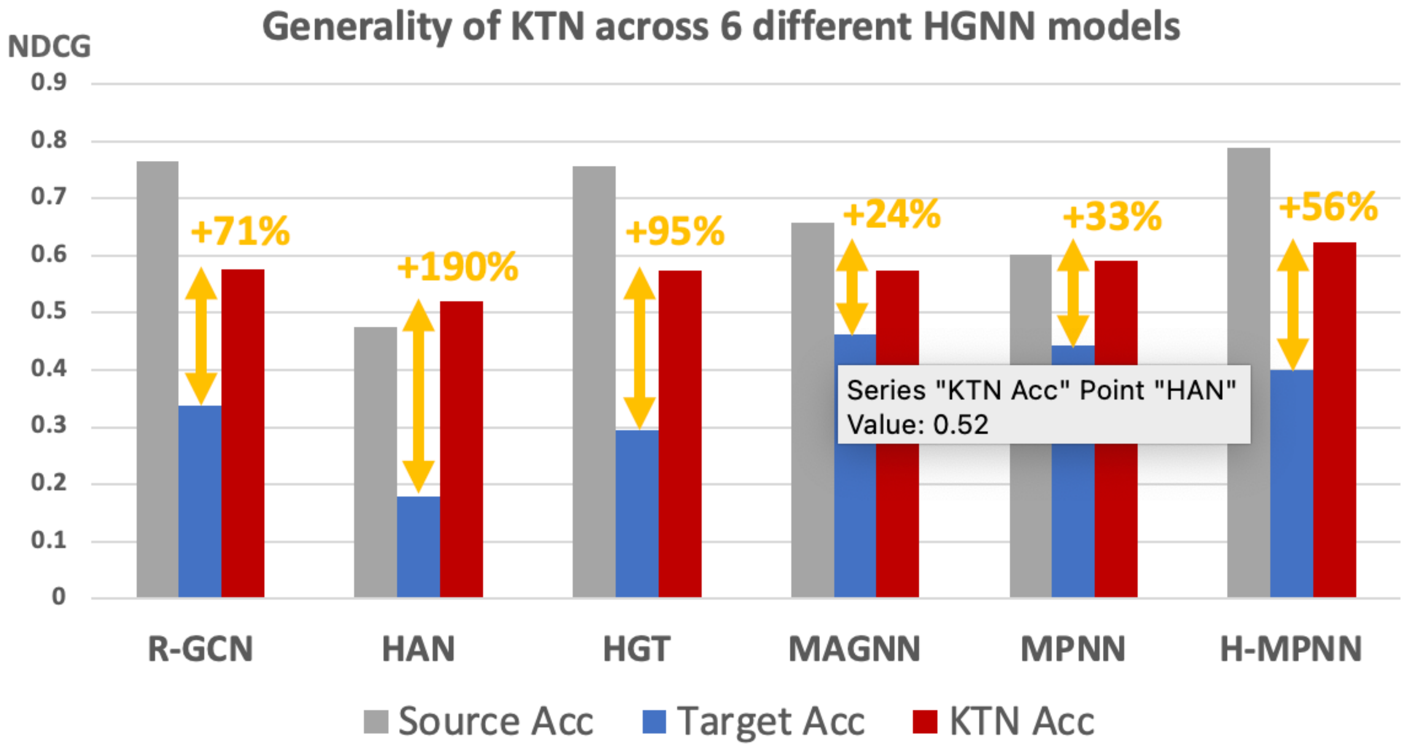




Yoon et al., Zero-shot Transfer Learning within a Heterogeneous Graph via Knowledge Transfer Networks, NeurIPS'22

Experiment (2) Generality

- 6 different HGNN models
- 960% improvement



Yoon et al., Zero-shot Transfer Learning within a Heterogeneous Graph via Knowledge Transfer Networks, NeurIPS'22

Paper: www.minjiyoon.xyz/Paper/KTN.pdf
Code: https://github.com/minjiyoon/KTN

NEURAL INFORMATION PROCESSING SYSTEMS

Check out our paper at NeurIPS 2022!

Zero-shot Transfer Learning within a Heterogeneous Graph via Knowledge Transfer Networks

Carnegie Mellon University

UCLA

Google Research