Self-supervised Auxiliary Learning with Meta-paths for Heterogeneous Graphs

Dasol Hwang*, Hyunwoo J. Kim etc
Korea University

NeurIPS 2020

0 Index

1. Introduction
2. Related
3. Proposed Approach
4. Experiments

Introduction: Why I choose this paper?

- **Hot Topic Self-supervised learning for GNN**
- **GNN for Real word heterogenous graph**
- **Meta Training**
- **For any GNN model & Performance Increase**

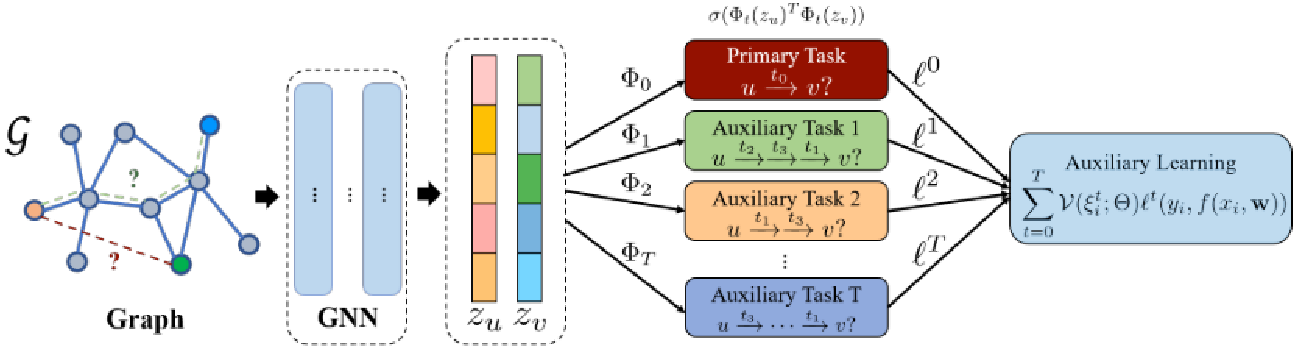
임진혁<eeplearning@units.ac.kr>

1 Introduction: What this paper proposes

Abstract

"We proposed meta-path prediction as self-supervised auxiliary tasks on heterogeneous graphs"

SELf-supervised Auxiliary LeaRning (SELAR).



Introduction: What this paper proposes

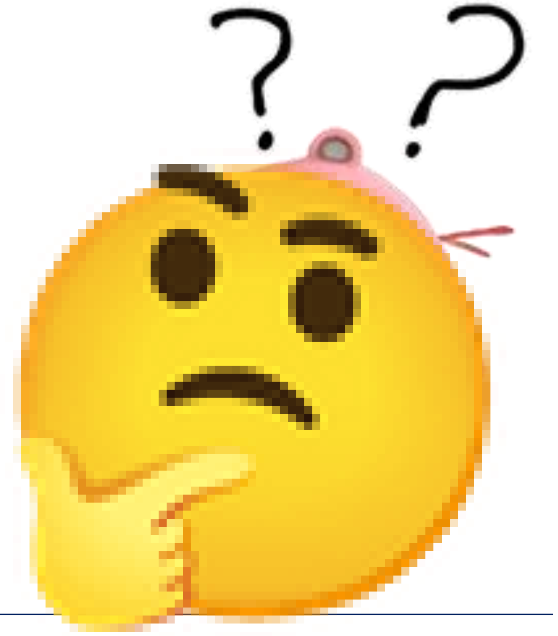
"We proposed meta-path prediction as self-supervised auxiliary tasks on heterogeneous graphs"

What is Heterogeneous Graphs**?**

What is Self-supervised leaning **?**

What is Auxiliary Learning **?**

What is Meta-Paths**?**



Related : Heterogeneous Graphs

Homogeneous Graphs: only one type node or edge

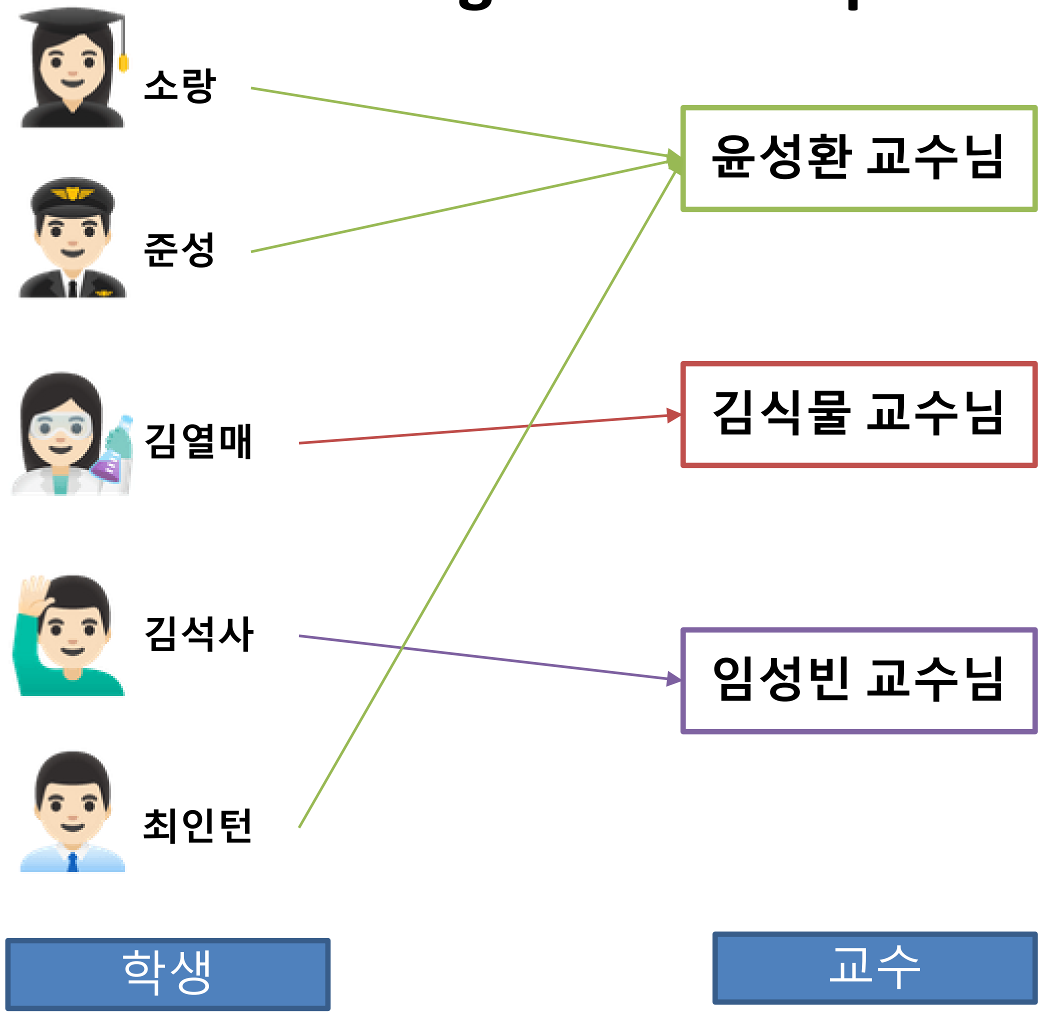
Heterogeneous Graphs: multiple type of node or edge

(like real-word)
(rich information for powerful representation)

임진혁<eeplearning@unist.ac.kr>

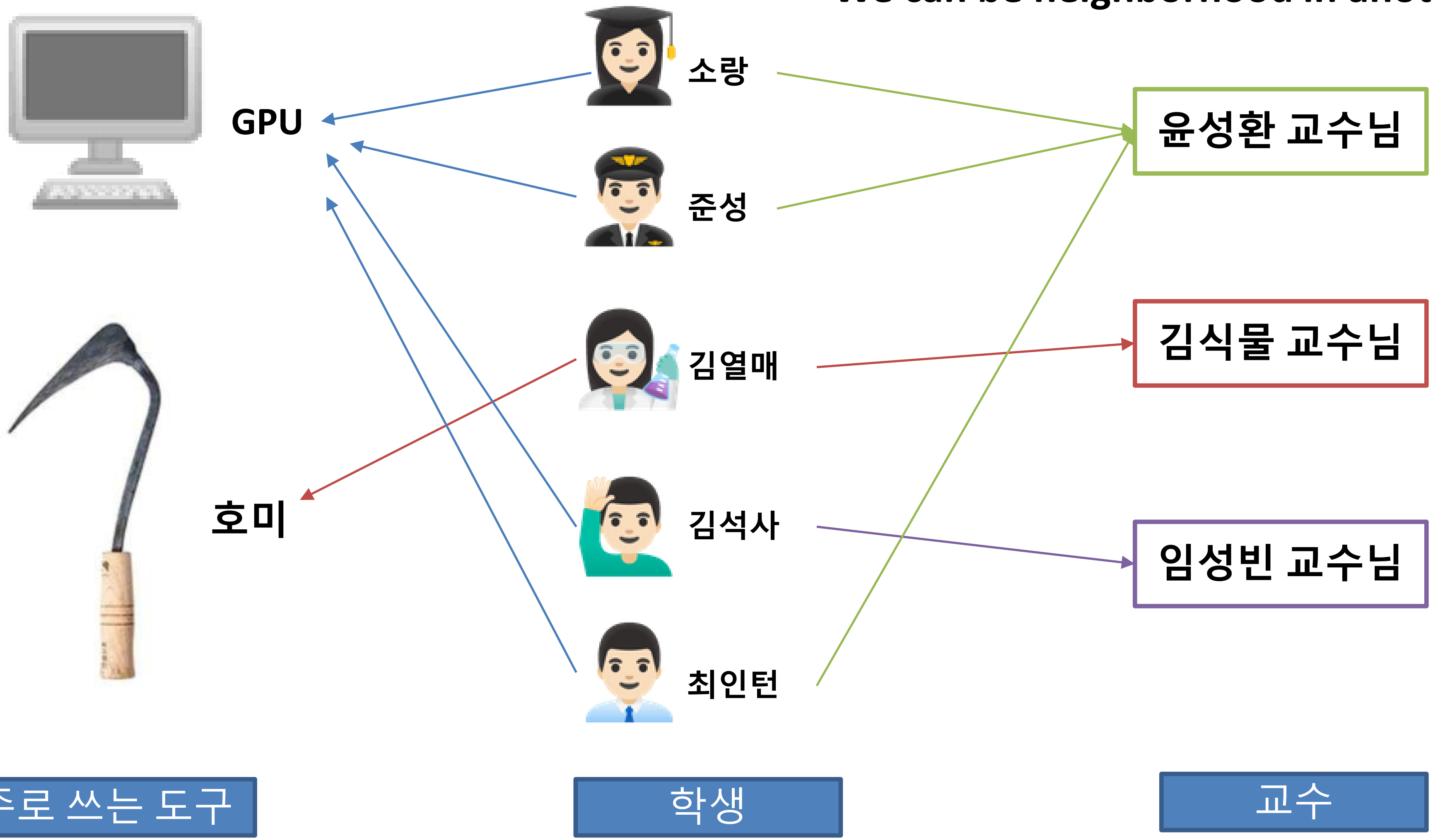
2 Related : Heterogeneous Graphs

Homogeneous Graphs can’t catch the rich information?

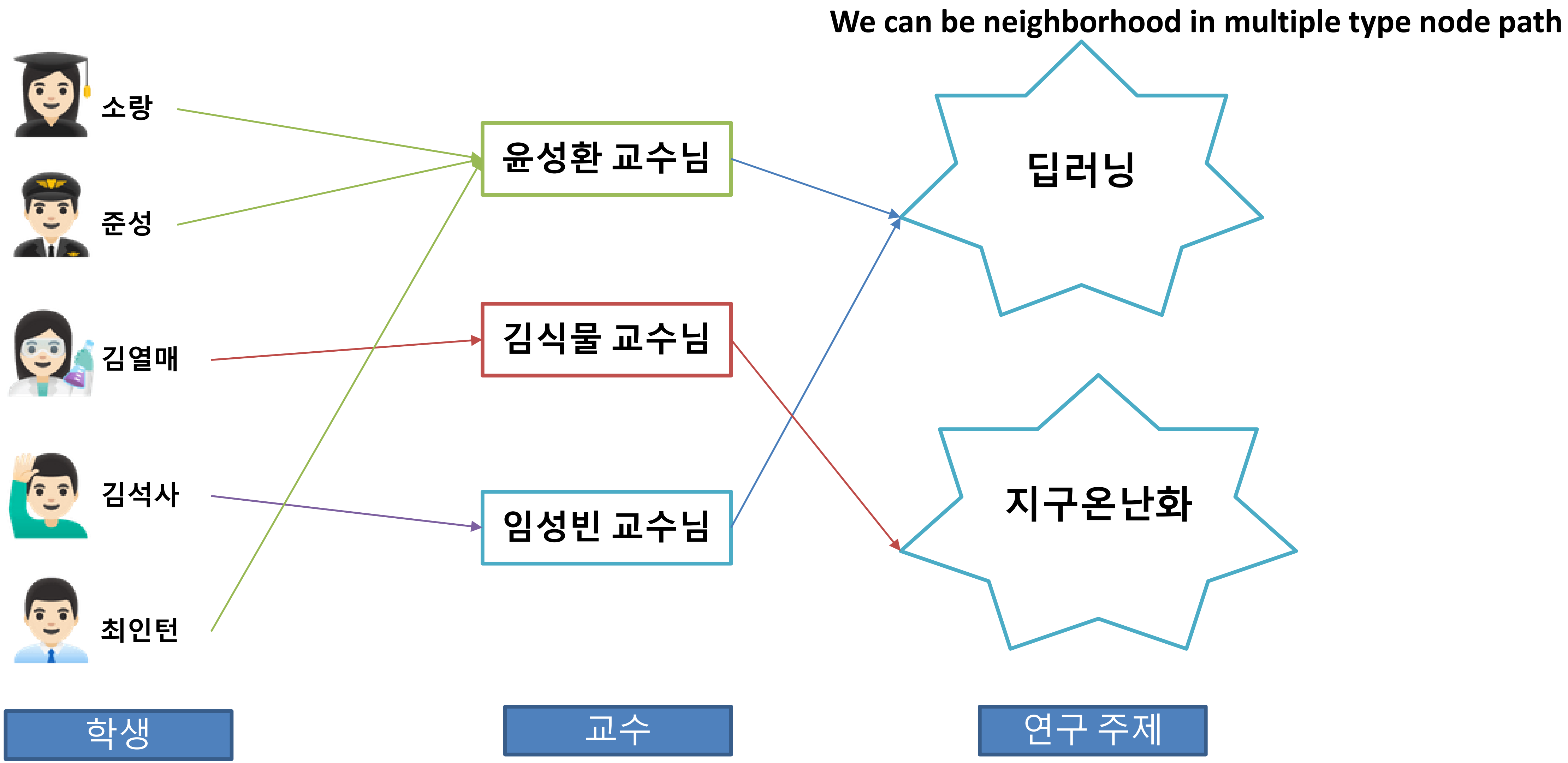


Related : Heterogeneous Graphs

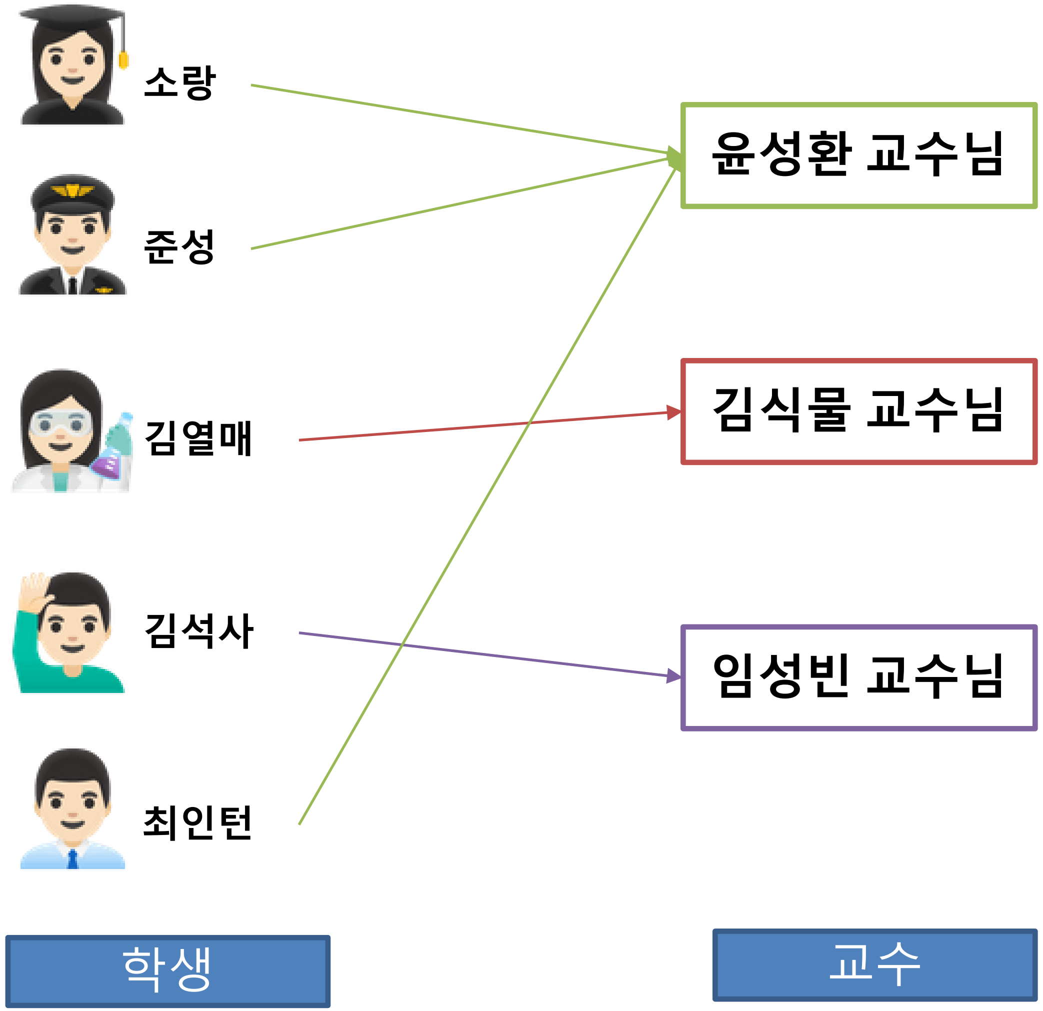
We can be neighborhood in another type node



Related : Heterogeneous Graphs



Homogeneous Graphs(multiple type node) have rich information!



임진혁<eeplearning@unist.ac.kr>

2 Related : Self-supervised learning

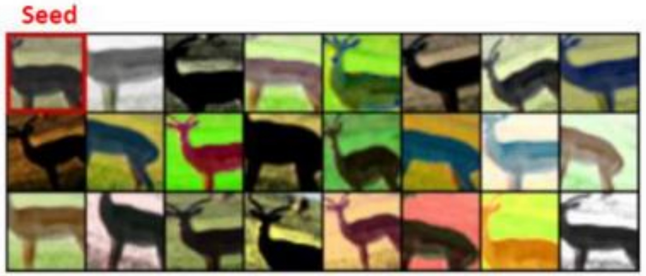
Self-Supervised Learning: **pretext task로 NN을 pretrain하여 downstream task로 transfer learning.**

pretext task : Unlabeled 데이터들을 이용하여 사용자가 새로운 문제를 정의하여 이에 대한 정답을
Self- supervised label이라 하며 이 때의 새로운 문제를 뜻함.

DOWNSTREAM TASK: **Pretrain된 가중치를 사용하여 원하는 테스크에 fine-tune**

self- supervised 학습은 일종의 비지도학습으로 라벨이 없는 데이터를 해당 데이터의 구조나 특성을 기반으로 라벨링하여 학습함으로써 High-level representations 학습을 가능케 한다.

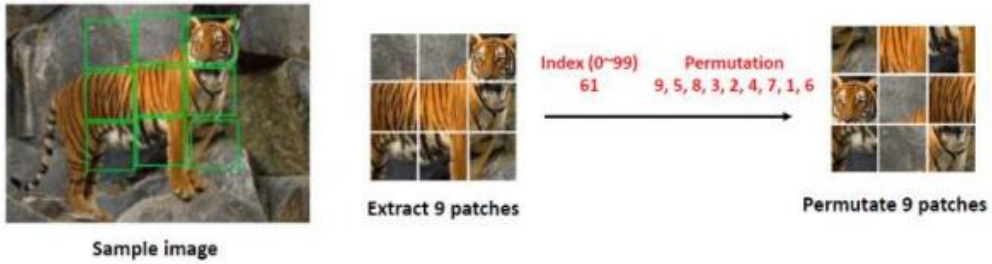
Exemplar, 2014 NIPS



Train with STL-10 dataset (96x96)

[Exemplar]

Jigsaw Puzzle, 2016 ECCV



[Jigsaw Puzzle]

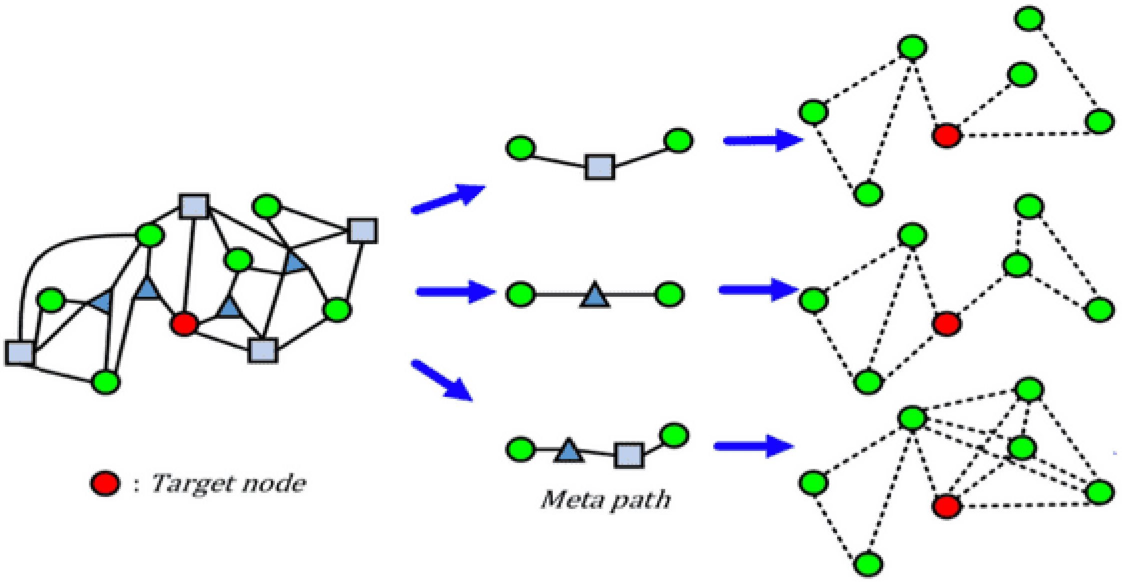
Related Concepts: Auxiliary Learning

employ auxiliary tasks to assist the primary task

Looks like multi-task learning BUT only care about the performance of the primary task

임진혁<eeplearning@unist.ac.kr>

2 Related Concepts: Meta-Paths



At Heterogeneous Graph Embedding

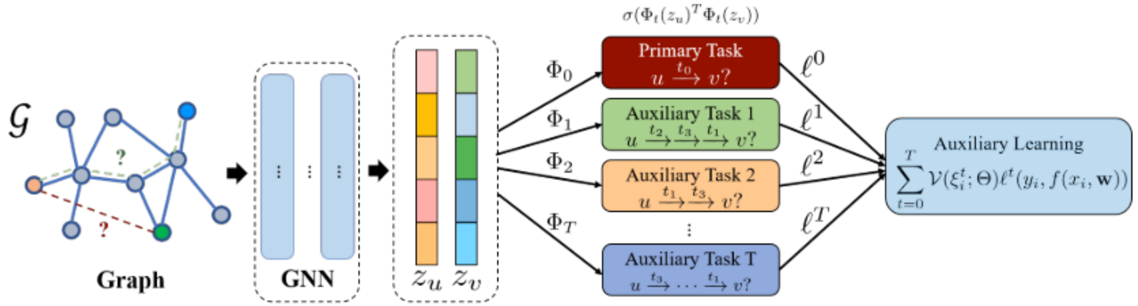
임진혁<eeplearning@unist.ac.kr>

Proposed Approach : What this paper proposes

Abstract

"We proposed meta-path prediction as self-supervised auxiliary tasks on heterogeneous graphs"

SELf-supervised Auxiliary LeaRning (SELAR).



Proposed Approach: **SELf-supervised Auxiliary LeaRning (SELAR).**

GOAL: learn with multiple auxiliary tasks to improve the performance of the primary task.

임진혁<eeplearning@unist.ac.kr>

Proposed Approach: **SELf-supervised Auxiliary LeaRning (SELAR).**

GOAL: learn with multiple auxiliary tasks to improve the performance of the primary task.

meta-path prediction

Hint Networks

임진혁<eeplearning@unist.ac.kr>

3 Proposed Approach: **SELf-supervised Auxiliary LeaRning (SELAR).**

GOAL: learn with multiple auxiliary tasks to improve the performance of the primary task.

$\Downarrow$

meta-path prediction $\subset$ Self-supervised leaning

임진혁<eeplearning@unist.ac.kr>

Proposed Approach: SELf-supervised Auxiliary LeaRning (SELAR).

GOAL: learn with multiple auxiliary tasks to improve the performance of the primary task.

⇓

meta-path prediction $\subset$ Self-supervised leaning **on** Heterogenous Graph

임진혁<eeplearning@unist.ac.kr>

3 Proposed Approach: SELf-supervised Auxiliary LeaRning (SELAR).

GOAL: learn with multiple auxiliary tasks to improve the performance of the primary task.

meta-path prediction ⊂ Self-supervised leaning **on** Heterogenous Graph

Understand Complicated and meaningful relations
Of Heterogenous Graph

임진혁<eeplearning@unist.ac.kr>

Proposed Approach: SELf-supervised Auxiliary LeaRning (SELAR).

Contribution: It isn’t just simple auxiliary learning for GNN

There are so many challenging problem at ***“Auxiliary task at GNN”***

- ***Graph Structure***
- ***Only Homogeneous Graph***
- ***Manually select the Auxiliary task with domain knowledge***

임진혁<eeplearning@unist.ac.kr>

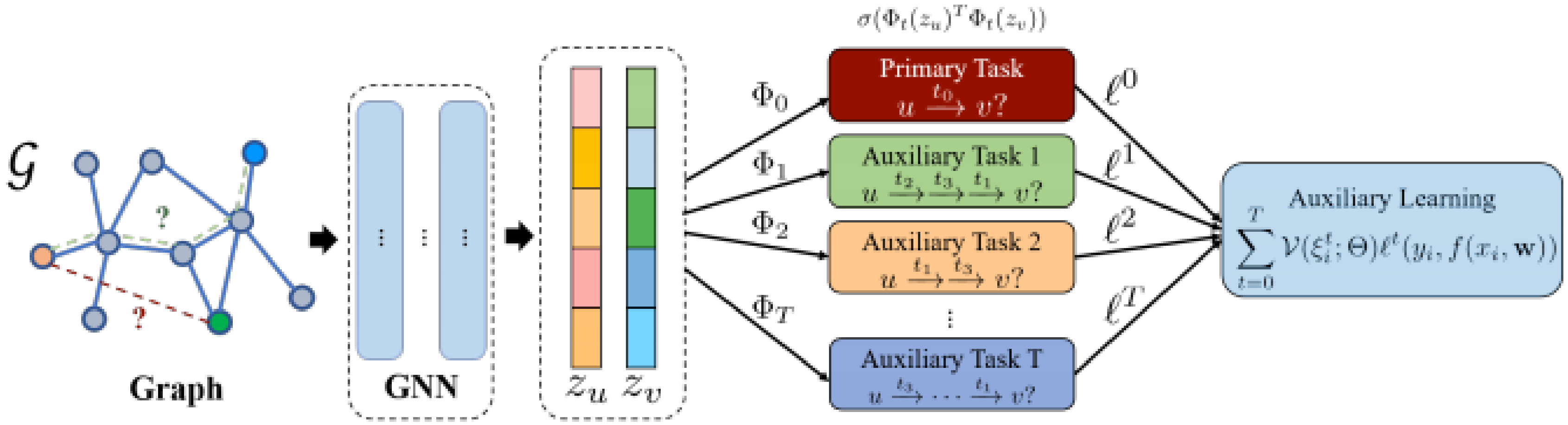
Proposed Approach: SELf-supervised Auxiliary LeaRning (SELAR)

Contribution: It isn't just simple auxiliary learning for GNN

- Propose a self-supervised learning method on a heterogeneous graph via meta-path prediction without additional data.
- Automatically selects meta-paths (auxiliary tasks) to assist the primary task via meta-learning.
- Develop Hint Network that helps the learner network to benefit from challenging auxiliary tasks.

임진혁<eeplearning@unist.ac.kr>

Proposed Approach: SELAR



1) learning weight functions to softly select auxiliary tasks and balance them with the primary task via meta-learning

2) learning Hint Networks to convert challenging auxiliary tasks into more relevant and solvable tasks to the primary task learner.

임진혁<eeplearning@unist.ac.kr>

Proposed Approach: SELAR

SELAR is learning to learn **a primary task** with multiple **auxiliary tasks** to assist the primary task

$$\min_{\mathbf{w},\Theta} \mathcal{L}^{pr}(\mathbf{w}^*(\Theta)) \quad \text{s.t. } \mathbf{w}^*(\Theta) = \underset{\mathbf{w}}{\operatorname{argmin}} \mathcal{L}^{pr+au}(\mathbf{w}; \Theta)$$

- $\mathcal{L}^{pr}$: Loss function for the primary task
- $\mathcal{L}^{pr+au}$: Loss functions for the primary task and auxiliary tasks
- $\mathbf{W}$: Model Parameters (for tasks)
- Θ : Parameters for meta-learning **(how to learn)**

임진혁<eeplearning@unist.ac.kr>

3 Proposed Approach: SELAR

SELAR is learning to learn **a primary task** with multiple **auxiliary tasks** to assist the primary task

$$\min_{\mathbf{w},\Theta} \mathcal{L}^{pr}(\mathbf{w}^*(\Theta)) \quad \text{s.t. } \mathbf{w}^*(\Theta) = \operatorname*{argmin}_{\mathbf{w}} \mathcal{L}^{pr+au}(\mathbf{w};\Theta)$$

Can be written as (just like my explanation)

$$\min_{\mathbf{w},\Theta} \sum_{i=1}^{M_0} \frac{1}{M_0} \ell^0(y_i^{(0,meta)}, f(x_i^{(0,meta)}; \mathbf{w}^*(\Theta))$$

$$\text{s.t. } \mathbf{w}^*(\Theta) = \operatorname*{argmin}_{\mathbf{w}} \sum_{t=0}^{T} \sum_{i=1}^{N_t} \frac{1}{N_t} \mathcal{V}(\xi_i^{(t,train)}; \Theta) \ell^t(y_i^{(t,train)}, f^t(x_i^{(t,train)}; \mathbf{w}))$$

- $\ell^t = \ell^t(y_i^{(t,train)}, f^t(x_i^{(t,train)}; \mathbf{w}))$. $\xi_i^{(t,train)}$
- $\xi_i^{(t,train)} = \left[\ell^t; e_t; y_i^{(t,train)}\right]$

임진혁<eeplearning@unist.ac.kr>

3 Proposed Approach: SELAR

SELAR is learning to learn **a primary task** with multiple **auxiliary tasks** to assist the primary task

$$\min_{\mathbf{w},\Theta} \mathcal{L}^{pr}(\mathbf{w}^*(\Theta)) \quad \text{s.t. } \mathbf{w}^*(\Theta) = \underset{\mathbf{w}}{\operatorname{argmin}}\, \mathcal{L}^{pr+au}(\mathbf{w};\Theta)$$

Can be written as (just like my explanation)

$$\min_{\mathbf{w},\Theta} \sum_{i=1}^{M_0} \frac{1}{M_0} \ell^0(y_i^{(0,meta)}, f(x_i^{(0,meta)}; \mathbf{w}^*(\Theta))$$

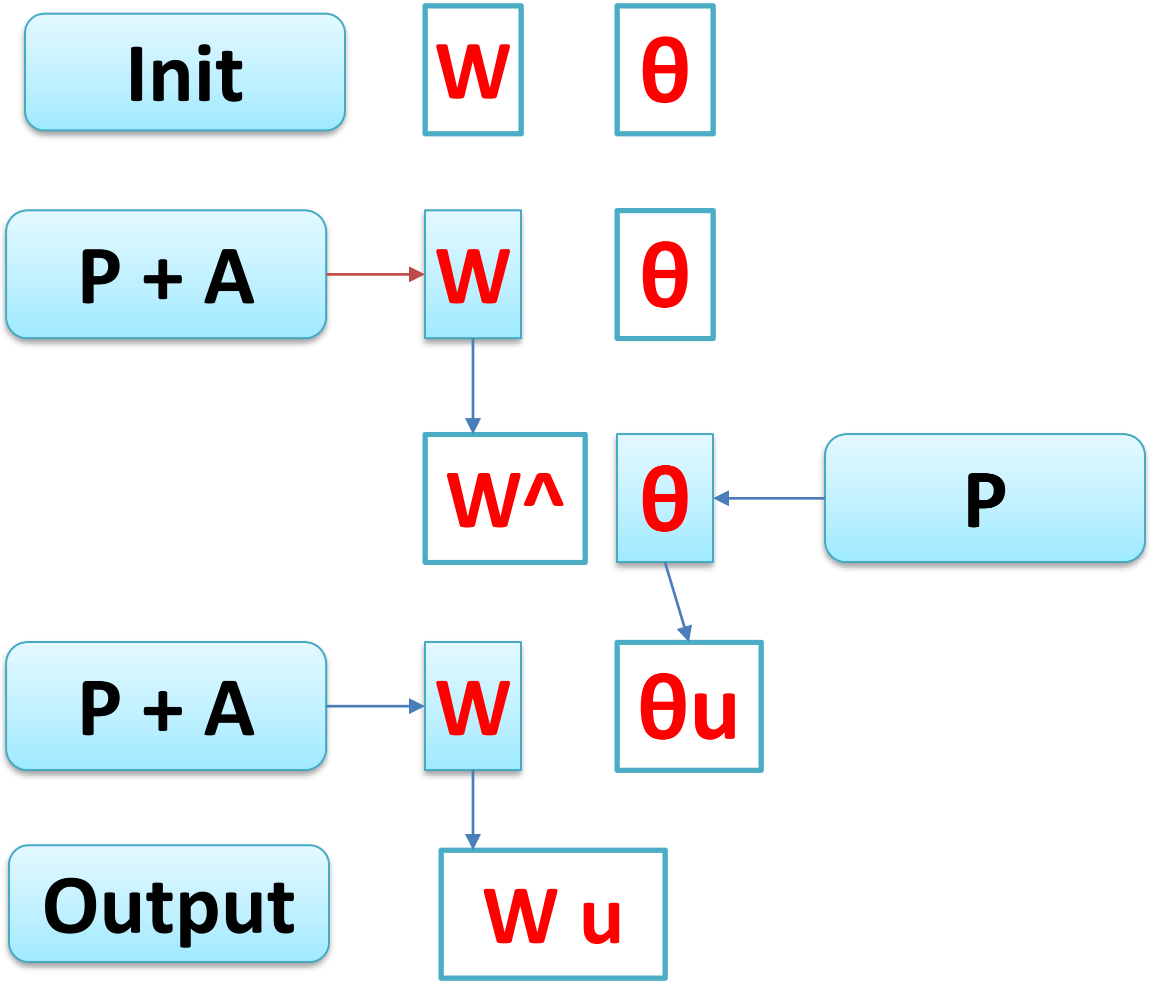
Loss function for task t

$$\text{s.t. } \mathbf{w}^*(\Theta) = \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{t=0}^{T} \sum_{i=1}^{N_t} \frac{1}{N_t} \mathcal{V}(\xi_i^t; \Theta) \cdot \ell^t(y_i, f(x_i; \mathbf{w}))$$

Weighting function

- $\ell^t = \ell^t(y_i^{(t,train)}, f^t(x_i^{(t,train)}; \mathbf{w}))$. $\xi_i^{(t,train)}$
- $\xi_i^{(t,train)} = \left[\ell^t; e_t; y_i^{(t,train)}\right]$

3 Proposed Approach: SELAR How to optimize this Bi-Optimization



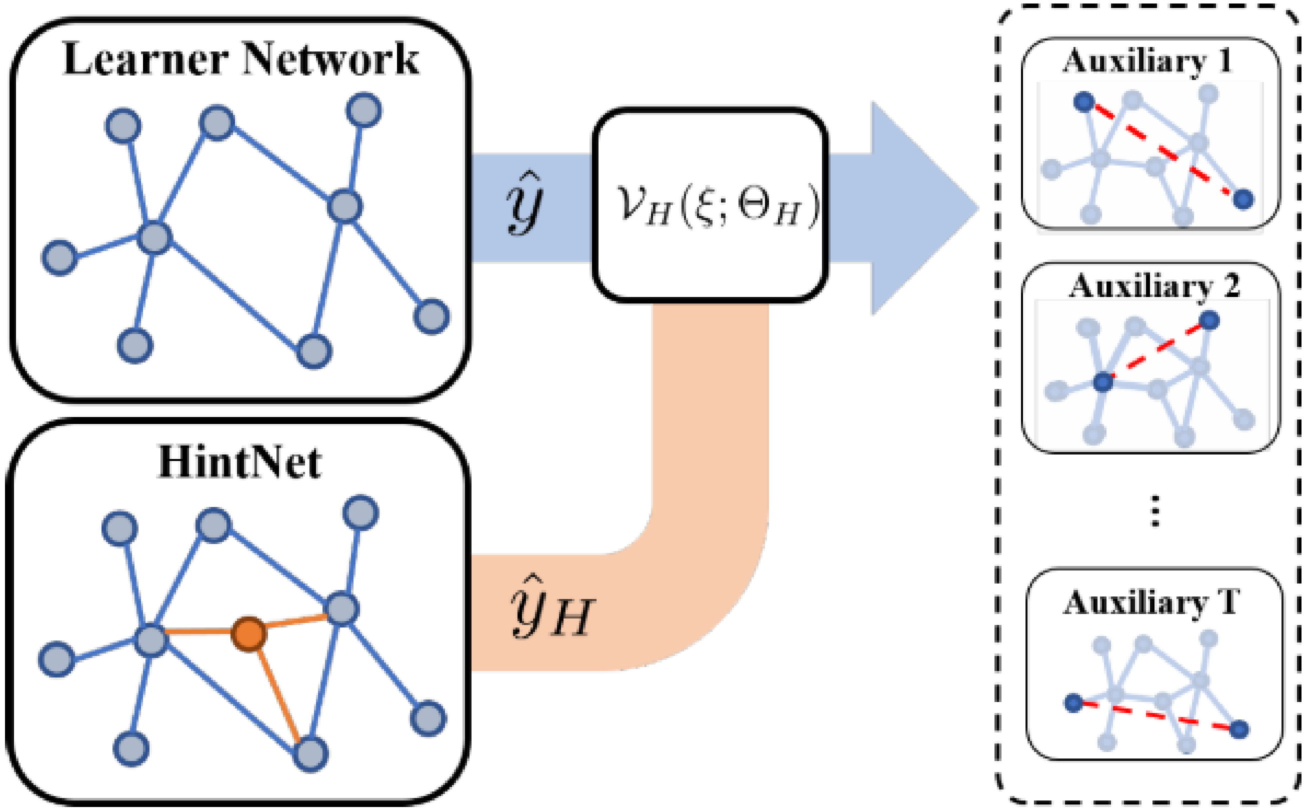
임진혁<eeplearning@unist.ac.kr>

Proposed Approach: Hint Network

Meta-path prediction more challeng problem than link prediction/ node classification

Provide Hint to convert meta path into more easy problem

convex combination of the learner's answer and HintNet's answer



임진혁<eeplearning@units.ac.kr>

Datasets

Link prediction : Last-FM and Book-Crossing with knowledge graph
Node classification : ACM and IMDB

Table 2: Datasets on heterogeneous graphs.

	Datasets	# Nodes	# Edges	# Edge type	# Features
Link prediction	Last-FM	15,084	73,382	122	N/A
	Book-Crossing	110,739	442,746	52	N/A
Node classification	ACM	8,994	25,922	4	1,902
	IMDB	12,772	37,288	4	1,256

임진혁<eeplearning@unist.ac.kr>

4 Experiments

Base Model: GCN , GAT , GIN, SG Conv and GTN

Q1. Is meta-path prediction effective for representation learning on **heterogeneous graphs**?

Q2. Can the meta-path prediction be further **improved** by the proposed methods?

Q3. Why are the **proposed methods** effective?

Table 1: **Link prediction** performance (AUC) of GNNs trained by various learning strategies.

Dataset	Base GNNs	Vanilla	w/o meta-path	w/ meta-path	Ours SELAR	SELAR+Hint
Last-FM	GCN	0.7963	0.7889	0.8235	**0.8296**	0.8121
	GAT	0.8115	0.8115	0.8263	0.8294	**0.8302**
	GIN	0.8199	0.8217	0.8242	**0.8361**	0.8350
	SGC	0.7703	0.7766	0.7718	0.7827	**0.7975**
	GTN	0.7836	0.7744	0.7865	0.7988	**0.8067**
	Avg. Gain	-	-0.0017	+0.0106	+0.0190	+0.0200
Book-Crossing	GCN	0.7039	0.7031	0.7110	0.7182	**0.7208**
	GAT	0.6891	0.6968	0.7075	0.7345	**0.7360**
	GIN	0.6979	0.7210	0.7338	**0.7526**	0.7513
	SGC	0.6860	0.6808	0.6792	0.6902	**0.6926**
	GTN	0.6732	0.6758	0.6724	**0.6858**	0.6850
	Avg. Gain	-	+0.0055	+0.0108	+0.0263	+0.0267

Table 2: **Node classification** performance ($F1$-score) of GNNs trained by various learning schemes.

Dataset	Base GNNs	Vanilla	w/o meta-path	w/ meta-path	Ours SELAR	SELAR+Hint
ACM	GCN	0.9091	0.9191	0.9104	0.9229	**0.9246**
	GAT	0.9161	0.9119	0.9262	0.9273	**0.9278**
	GIN	0.9085	0.9118	0.9058	0.9092	**0.9135**
	SGC	0.9163	0.9194	0.9223	0.9224	**0.9235**
	GTN	0.9181	0.9191	0.9246	**0.9258**	0.9236
	Avg. Gain	-	+0.0027	+0.0043	+0.0079	**+0.0090**
IMDB	GCN	0.5767	0.5855	0.5994	0.6083	**0.6154**
	GAT	0.5653	0.5488	0.5910	**0.6099**	0.6044
	GIN	0.5888	0.5698	0.5891	**0.5931**	0.5897
	SGC	0.5779	0.5924	0.5940	0.6151	**0.6192**
	GTN	0.5804	0.5792	0.5818	0.5994	**0.6063**
	Avg. Gain	-	-0.0027	+0.0132	+0.0274	**+0.0292**

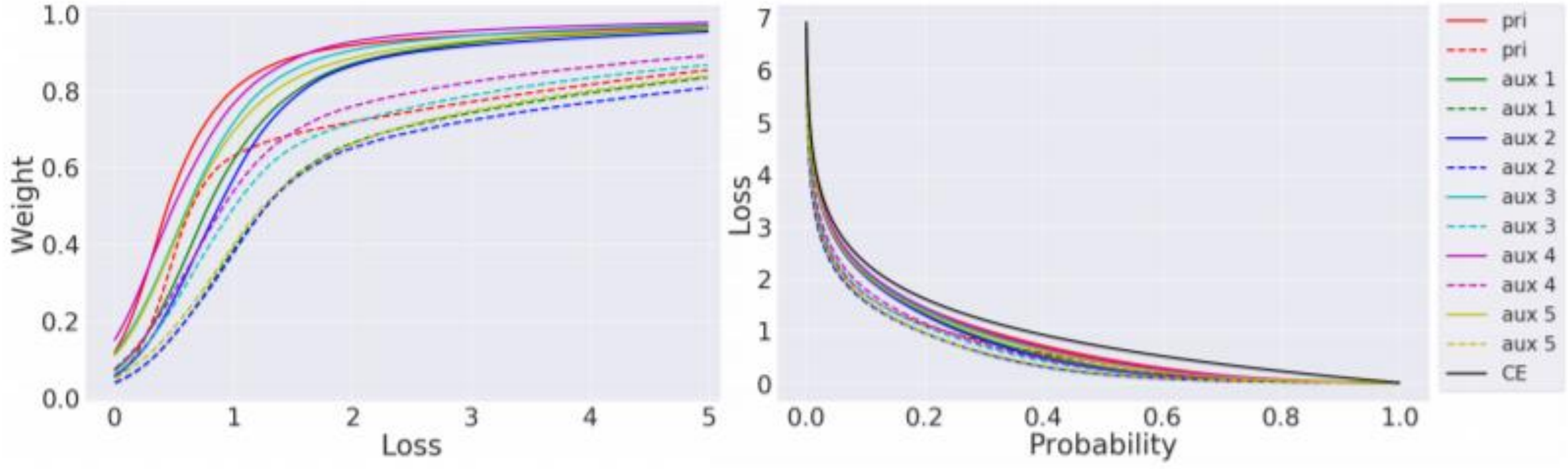
임진혁<eeplearning@unist.ac.kr>

Experiments

Q3. Why are the **proposed methods** effective?

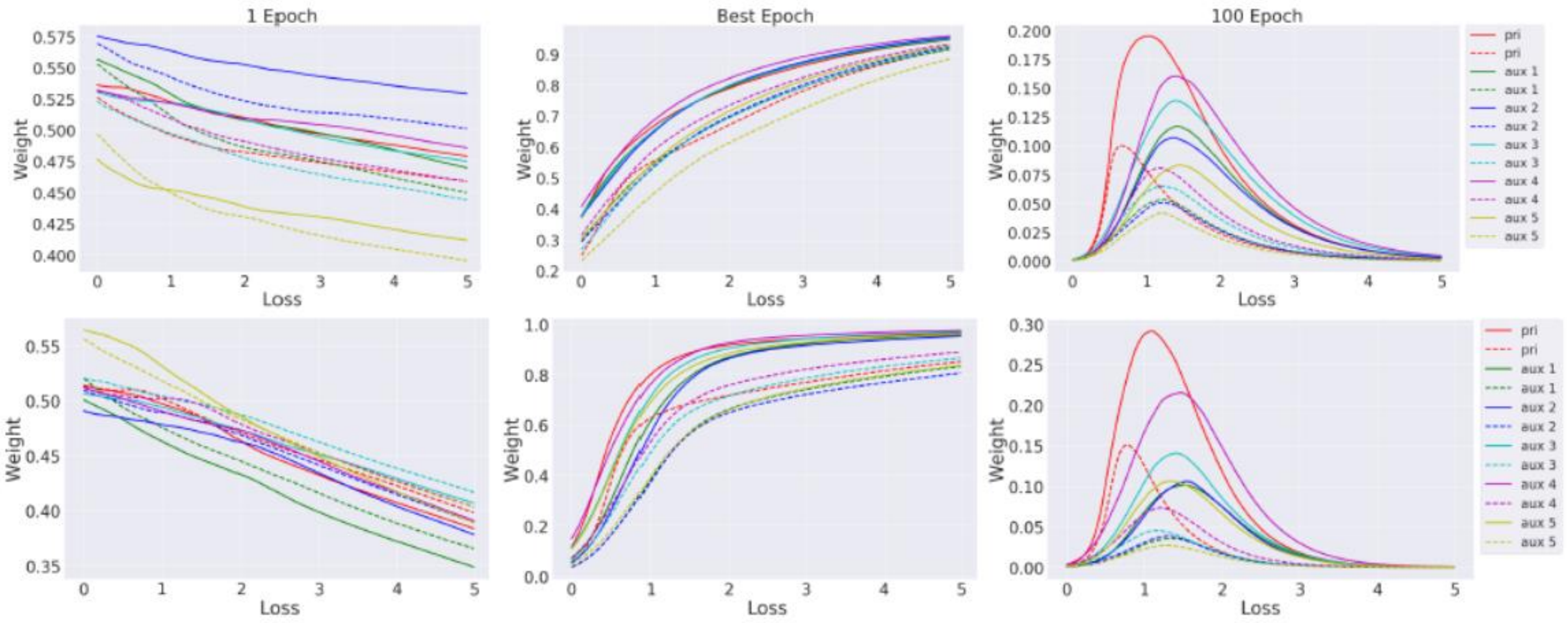
How we know that?

Work like **Focal loss : focus on hard samples(task)**



(a) Weighting function $\mathcal{V}(\xi;\Theta)$. (b) Adjusted Cross Entropy $\mathcal{V}(\xi;\Theta)\ell^{t}(y,\hat{y})$.

Q3. Why are the **proposed methods** effective?



4 Experiments

Q3. Why are the **proposed methods** effective?

Table 2: The average of the task-specific weighted loss on **Last-FM** and **Book-Crossing** datasets.

Meta-paths (Last-FM)	Avg.	Meta-paths (Book-Crossing)	Avg.
user-item-actor-item	**7.675**	user-item*	**6.439**
user-item*	7.608	user-item-literary.series-item-user	6.217
user-item-appearing.in.film-item	7.372	item-genre-item	6.163
user-item-instruments-item	7.049	user-item-user-item	6.126
user-item-user-item	6.878	user-item-user	6.066
user-item-artist.origin-item	6.727	item-user-item	6.025

∗ primary task

4 Experiments

Meta cross-validation

Table 3: Comparison between 1-fold and 3-fold as meta-data on **Last-FM** datasets.

Model	Vanilla	SELAR 1-fold	SELAR 3-fold	SELAR+Hint 1-fold	SELAR+Hint 3-fold
GCN	0.7963	0.7885	**0.8296**	0.7834	**0.8121**
GAT	0.8115	0.8287	**0.8294**	0.8290	**0.8302**
GIN	0.8199	0.8234	**0.8361**	0.8244	**0.8350**
SGC	0.7703	0.7691	**0.7827**	0.7702	**0.7975**
GTN	0.7836	0.7897	**0.7988**	0.7915	**0.8067**

Algorithm 1 Self-supervised Auxiliary Learning

Input: training data for primary/auxiliary tasks D^{pr}, D^{au}, mini-batch size N_{pr}, N_{au}
Input: max iterations K, # folds for cross validation C, learning rate α, β
Output: network parameter $\mathbf{w}^K$ for the primary task

```
Initialize w^1, Θ^1
for k = 1 to K do
    D_m^pr ← MiniBatchSampler(D^pr, N_pr)
    D_m^au ← MiniBatchSampler(D^au, N_au)
    for c = 1 to C do                                              ▷ Meta Learning with Cross Validation
        D_m^pr(train), D_m^pr(meta) ← CVSplit(D_m^pr, c)            ▷ Split Data for CV
        ŵ^k(Θ^k) ← w^k − α∇_w L^{pr+au}(w^k; Θ^k) with D_m^pr(train) ∪ D_m^au   ▷ Eq. (6)
        g_c ← ∇_Θ L^pr(ŵ^k(Θ^k)) with D_m^pr(meta)                  ▷ Eq. (7)
    end for
    Update Θ^{k+1} ← Θ^k − β Σ_c^C g_c                              ▷ Eq. (9)
    w^{k+1} = w^k − α∇_w L^{pr+au}(w^k; Θ^{k+1}) with D_m^pr ∪ D_m^au        ▷ Eq. (8)
end for
```


임진혁<eeplearning@unist.ac.kr>

Question about the result

- **Hint Network really work?**
- **How it will be apply Hint Network to “with meta” model**
- **Can MetaCV be applied to any Meta?**

Table 1: **Link prediction** performance (AUC) of GNNs trained by various learning strategies.

Dataset	Base GNNs	Vanilla	w/o meta-path	w/ meta-path	Ours SELAR	SELAR+Hint
Last-FM	GCN	0.7963	0.7889	0.8235	**0.8296**	0.8121
	GAT	0.8115	0.8115	0.8263	0.8294	**0.8302**
	GIN	0.8199	0.8217	0.8242	**0.8361**	0.8350
	SGC	0.7703	0.7766	0.7718	0.7827	**0.7975**
	GTN	0.7836	0.7744	0.7865	0.7988	**0.8067**
	Avg. Gain	-	-0.0017	+0.0106	+0.0190	+0.0200
Book-Crossing	GCN	0.7039	0.7031	0.7110	0.7182	**0.7208**
	GAT	0.6891	0.6968	0.7075	0.7345	**0.7360**
	GIN	0.6979	0.7210	0.7338	**0.7526**	0.7513
	SGC	0.6860	0.6808	0.6792	0.6902	**0.6926**
	GTN	0.6732	0.6758	0.6724	**0.6858**	0.6850
	Avg. Gain	-	+0.0055	+0.0108	+0.0263	+0.0267

Table 2: **Node classification** performance ($F1$-score) of GNNs trained by various learning schemes.

Dataset	Base GNNs	Vanilla	w/o meta-path	w/ meta-path	Ours SELAR	SELAR+Hint
ACM	GCN	0.9091	0.9191	0.9104	0.9229	**0.9246**
	GAT	0.9161	0.9119	0.9262	0.9273	**0.9278**
	GIN	0.9085	0.9118	0.9058	0.9092	**0.9135**
	SGC	0.9163	0.9194	0.9223	0.9224	**0.9235**
	GTN	0.9181	0.9191	0.9246	**0.9258**	0.9236
	Avg. Gain	-	+0.0027	+0.0043	+0.0079	**+0.0090**
IMDB	GCN	0.5767	0.5855	0.5994	0.6083	**0.6154**
	GAT	0.5653	0.5488	0.5910	**0.6099**	0.6044
	GIN	0.5888	0.5698	0.5891	**0.5931**	0.5897
	SGC	0.5779	0.5924	0.5940	0.6151	**0.6192**
	GTN	0.5804	0.5792	0.5818	0.5994	**0.6063**
	Avg. Gain	-	-0.0027	+0.0132	+0.0274	**+0.0292**

감사합니다

임진혁<eeplearning@unist.ac.kr>