Shift-Robust GNNs: Overcoming the Limitations of Localized Graph Training Data

Presented by Yian Wang

10/27

ILLINOIS

Outline

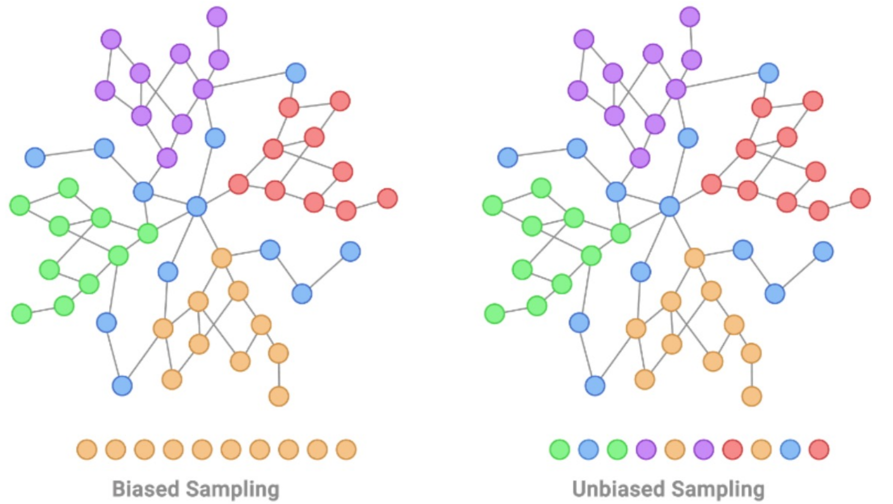
- Motivation
- Preliminaries
- Shift-Robust GNN
- Experiments
- Future work

Outline

- Motivation
- Preliminaries
- Shift-Robust GNN
- Experiments
- Future work

IID v.s. Localized training data

- IID assumption
- Biased training data
  - Spam and abuse detection problem
  - Human annotations
    - Sparse
    - Biased (< 1% positive)



[1] Liu, Z., Chen, C., Yang, X., Zhou, J., Li, X., & Song, L. (2018, October). Heterogeneous graph neural networks for malicious account detection. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (pp. 2077-2085).
[2] Halcrow, J., Mosoi, A., Ruth, S., & Perozzi, B. (2020, August). Grale: Designing networks for graph learning. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2523-2532).

Outline

- Motivation
- Preliminaries
- Shift-Robust GNN
- Experiments
- Future work

Preliminaries

- GNN
  - Given:
    - A graph $G = \{V, E\}$
    - Feature matrix $X \in \mathbb{R}^{|V| \times F}$
    - Adjacency matrix $A \in \mathbb{R}^{|V| \times |V|}$
  - Output:
    - Node representations Z
    - Node labels $\{y_i\}$

[1] Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Preliminaries

- GNN
  - A general GNN layer, final representation $\mathrm{Z} = \mathrm{H}^{k}$

$$H^{k} = \sigma(\tilde{A}H^{k-1}\theta^{k})$$

  - For node classification, a cross-entropy loss function is used

$$\mathcal{L} = \frac{1}{M}\sum_{i=1}^{M} l(y_i, z_i)$$

[1] Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Preliminaries

- Distribution shifts in GNN
  - $P_{train}(X, Y) \neq P_{test}(X, Y)$
  - In a neural network, only care about shift in the last hidden activated layer Z, i.e. $P_{train}(Z, Y) \neq P_{test}(Z, Y)$
  - Assume $P_{train}(Y|Z) = P_{test}(Y|Z)$, s.t.

$$P_{train}(Z) \neq P_{test}(Z)$$

[1] Quinonero-Candela, J., Sugiyama, M., Schwaighofer, A., & Lawrence, N. D. (Eds.). (2008). *Dataset shift in machine learning*. Mit Press.

Preliminaries

- General solutions to distribution shift
  - Importance weighting

$$\hat{\theta} = \operatorname{argmin}_{\theta\in\Theta} \frac{1}{M} \sum_{i=1}^{M} l(y_i, \hat{y}_i)$$

$$\rightarrow \hat{\theta} = \operatorname*{argmin}_{\theta\in\Theta} \frac{1}{M} \sum_{i=1}^{M} \frac{P_{test}(x_i)}{P_{train}(x_i)} l(y_i, \hat{y}_i)$$

  - Regularizations using discrepancy measures
    - Maximum mean discrepancy (MMD)

$$\mathrm{MMD} = \| E_p\phi(X) - E_q\phi(Y) \|_{H_k}^2, k(X,Y) = < \phi(X), \phi(Y) >$$

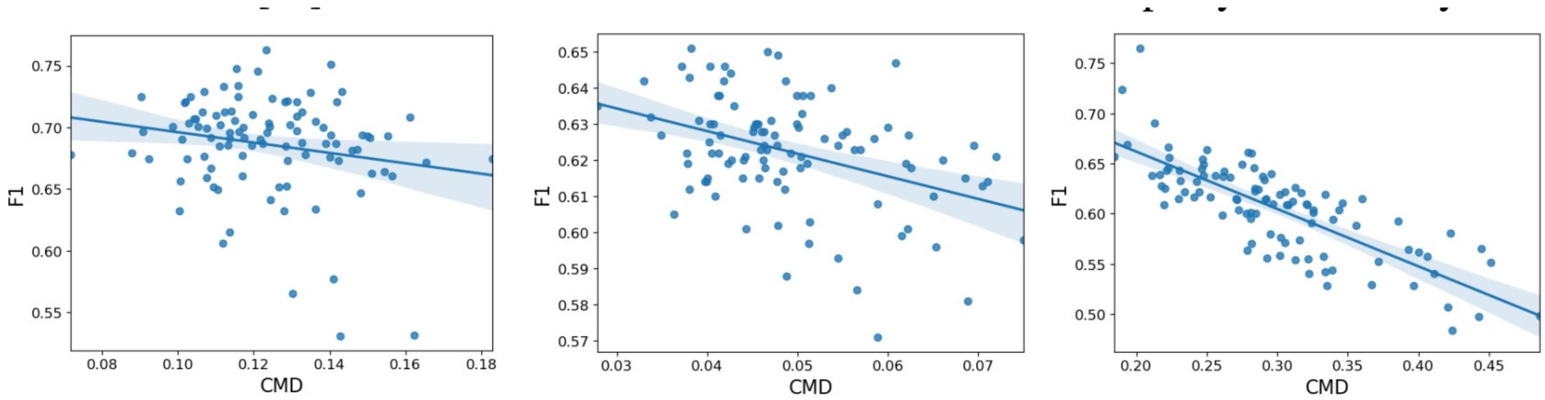
    - Central moment discrepancy (CMD)

$$\mathrm{CMD} = \frac{1}{|b-a|} \| E(p) - E(q) \|_2 + \sum_{k=2}^{\infty} \frac{1}{|b-a|^k} \| c_k(p) - c_k(q) \|_2$$

[1] Sugiyama, M., Krauledat, M., & Müller, K. R. (2007). Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, *8*(5).
[2] Long, M., Cao, Y., Wang, J., & Jordan, M. (2015, June). Learning transferable features with deep adaptation networks. In *International conference on machine learning* (pp. 97-105). PMLR.
[3] Zellinger, W., Moser, B. A., Grubinger, T., Lughofer, E., Natschläger, T., & Saminger-Platz, S. (2019). Robust unsupervised domain adaptation for neural networks via moment alignment. *Information Sciences*, *483*, 174-191.

Preliminaries

- Effect of distribution shifts



(a) Cora **(b)** Citeseer **(c)** Pubmed

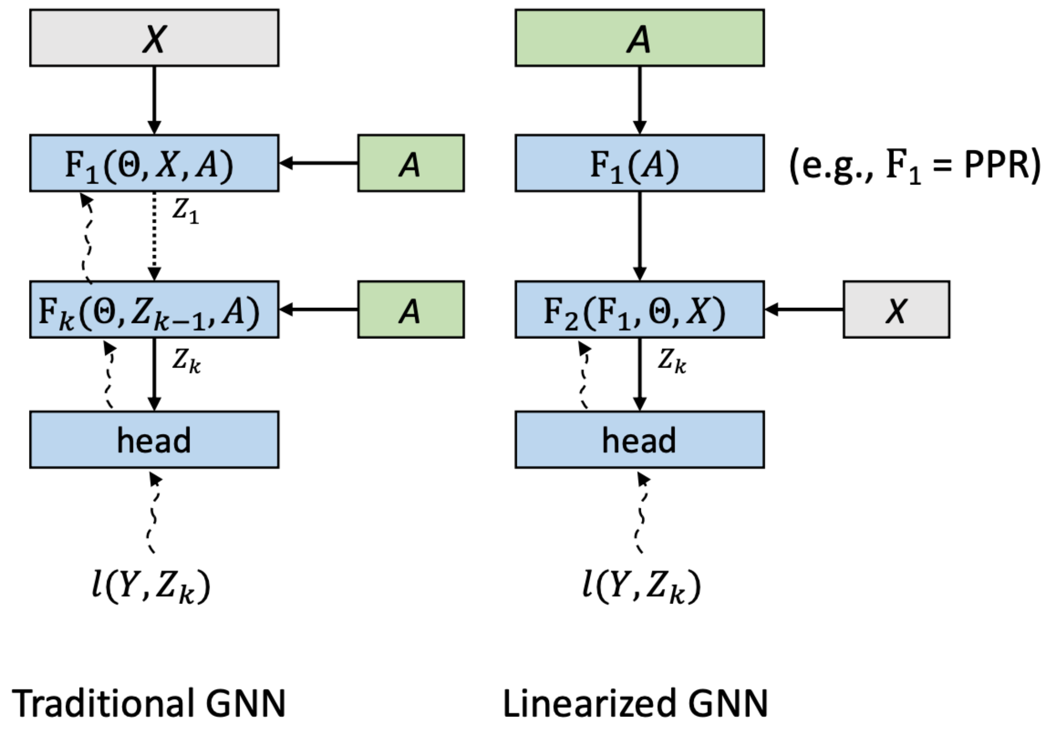
The effect of distribution shift on different datasets.
Y-axis: F1 score
X-axis: CMD

Outline

- Motivation
- Preliminaries
- Shift-Robust GNN
  - Standard GNN models
  - Linearized GNN models
- Experiments
- Future work

Shift Robust GNN

- Standard & Linearized GNN



Shift Robust GNN

- Standard GNN

$$\Phi = F(\Theta, A)$$

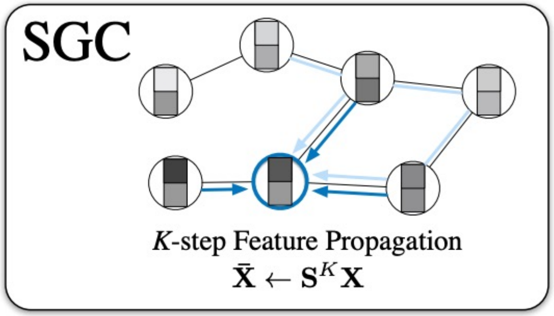
- Sample an IID data of the same size of training data and minimize the distribution between Z_{train} and Z_{IID}
  - $\mathcal{L} = \frac{1}{M}\sum_{i=1}^{M} l(y_i, z_i) + \lambda d(Z_{train}, Z_{IID})$
  - $\mathrm{d} = \frac{1}{|b-a|} \| E(Z_{train}) - E(Z_{IID}) \|_2 + \sum_{k=2}^{\infty} \frac{1}{|b-a|^k} \| c_k(Z_{train}) - c_k(Z_{IID}) \|_2$

Shift Robust GNN

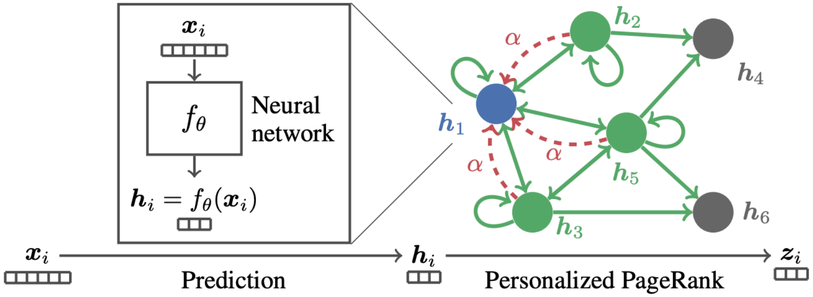
- Linearized GNN

$$\Phi = F_2(F_1(A), \Theta, X)$$

$$\mathcal{L} = \frac{1}{M}\sum_{i=1}^{M} \beta_i l(y_i, \Phi(h_i))$$



$$SGC\colon \mathrm{F}_1(A) = A^k X$$



$$APPNP\colon \mathrm{F}_1(A) = \alpha\left(I - (1-\alpha)\tilde{A}\right)^{-1}$$

$$H^0 = Z^0 = f_\theta(X)$$

$$H^{k+1} = (1-\alpha)\tilde{A}H^k + \alpha H^0$$

$$H^k = softmax((1-\alpha)\tilde{A}H^k + \alpha H^0)$$

[1] Wu, F., Souza, A., Zhang, T., Fifty, C., Yu, T., & Weinberger, K. (2019, May). Simplifying graph convolutional networks. In *International conference on machine learning* (pp. 6861-6871). PMLR.
[2] Klicpera, J., Bojchevski, A., & Günnemann, S. (2018). Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997*.

Shift Robust GNN

- Linearized GNN
  - Use importance sampling and calculate the instance weight via kernel mean matching (KMM):

$$\min_{\beta_i} \| \frac{1}{M}\sum_{i=1}^{M} \beta_i \psi(h_i) - \frac{1}{M'}\sum_{i=1}^{M'} \psi(h_i') \|^2 \;\; s.t. B_l \le \beta \le B_u$$

$\{h_i\}_{i=1}^{M}$: Biased training sample

$\{h_i'\}_{i=1}^{M}$: IID sample

ψ: $\mathbb{R}^{\mathrm{n}} \to \mathcal{H}$: Feature map to the reproducing kernel Hilbert space

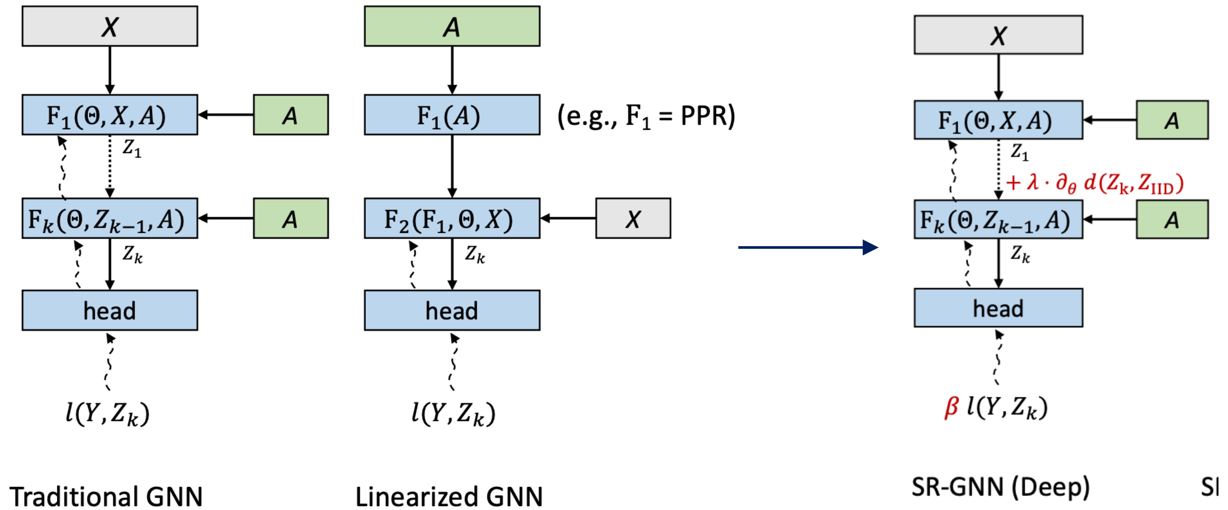
[1] Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., & Schölkopf, B. (2009). Covariate shift by kernel mean matching. *Dataset shift in machine learning*, *3*(4), 5.

Shift Robust GNN Framework

$$\mathcal{L} = \frac{1}{M}\sum_{i=1}^{M} \beta_i l(y_i, \Phi(h_i)) + \lambda d(Z_{train}, Z_{IID})$$

- Where for Φ, choose APPNP as a concrete instance.

$$\Phi_{APPNP} = ((1-\alpha)^k \tilde{A}^k + \alpha \sum_{i=0}^{k-1} (1-\alpha)^i \tilde{A}^i) F(\Theta, X)$$



[1] Klicpera, J., Bojchevski, A., & Günnemann, S. (2018). Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997*.

Outline

- Motivation
- Preliminaries
- Shift-Robust GNN
- Experiments
  - main result
  - Parameter sensitivity
- Future work

Experiments

- Biased Training Set Creation
  - Need an efficient method for finding 'nearby' nodes in the graph for a particular seed node.

Algorithm 1: Biased Training Set Creation PPR-S(γ, ϵ, α)

Given a class c, label ratio τ, graph size N;
Initialize the biased training set X = { } ;
while *len(X)*$< N \cdot \tau$ **do**
    Sample node i of class c, compute its top-γ entries in $\pi_i^{\text{ppr}}(\epsilon)$ via [2];
    if $\pi_i^{ppr}(\epsilon)$ *has* γ *non-zero entries* **then**
        X.add($\pi_i^{\text{ppr}}(\epsilon)$) ;
    end
end

$$\pi_i^{ppr} = (1-\alpha)\tilde{A}\pi_i^{ppr} + \alpha H^0$$

[1] Andersen, R., Chung, F., & Lang, K. (2006, October). Local graph partitioning using pagerank vectors. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)* (pp. 475-486). IEEE.

Experiments

(a) IID sample **(b)** Biased sample **(c)** PPR-score on biased sample

Figure 5: A biased sample on Cora dataset for one class, orange indicates the training data, red indicates the initial seed used in our PPR-S sampler. The PPR-score is presented in figure (c).

Experiments

- Datasets

Table 5: Overall Dataset Statistics

Dataset	# Nodes	# Edges	# Features	# Classes
Cora	2,708	5,278	1,433	7
Citeseer	3,327	4,614	3,703	6
PubMed	19,717	44,325	500	3
ogb-arxiv	169,343	1,166,243	128	40
Reddit	232,965	114,615,892	602	41

Experiments

- Experimental results on small benchmarks

Table 1: Semi-supervised classification on three different citation networks using biased training samples. Our proposed framework (SR-GNN) outperforms **all** baselines on biased training input.

Method	Cora			Citeseer			PubMed		
	Micro-F1↑	Macro-F1↑	ΔF1 ↓	Micro-F1↑	Macro-F1↑	ΔF1 ↓	Micro-F1↑	Macro-F1↑	ΔF1 ↓
GCN (IID)	80.8 ± 1.6	80.1 ± 1.3	0	70.3 ± 1.9	66.8 ± 1.3	0	79.8 ± 1.4	78.8 ± 1.4	0
Feat.+MLP	49.7 ± 2.5	48.3 ± 2.2	31.1	55.1 ± 1.3	52.7 ± 1.3	25.2	51.3 ± 2.8	41.8 ± 6.2	28.5
Emb.+MLP	57.6 ± 3.0	56.2 ± 3.0	23.2	38.5 ± 1.2	38.6 ± 1.1	31.8	60.4 ± 2.1	56.6 ± 2.0	19.4
DGI	71.7 ± 4.2	69.2 ± 3.7	9.1	62.6 ± 1.6	60.0 ± 1.6	7.6	58.0 ± 5.3	52.4 ± 8.3	21.8
GCN	67.6 ± 3.5	66.4 ± 3.0	13.2	62.7 ± 1.8	60.4 ± 1.6	7.6	60.6 ± 3.8	56.0 ± 6.0	19.2
GAT	58.4 ± 5.7	58.5 ± 5.0	22.4	58.0 ± 3.5	55.0 ± 2.7	12.3	55.2 ± 3.7	46.0 ± 6.4	14.6
SGC	70.2 ± 3.0	68.0 ± 3.8	10.6	65.4 ± 0.8	62.5 ± 0.8	4.9	61.8 ± 4.5	57.4 ± 7.2	18.0
APPNP	71.3 ± 4.1	69.2 ± 3.4	9.5	63.4 ± 1.8	61.2 ± 1.6	6.9	63.4 ± 4.2	58.7 ± 7.0	16.4
SR-GNN **w.o.** IR	72.1 ± 4.4	69.8 ± 3.7	8.7	63.9 ± 0.7	61.8 ± 0.6	6.4	69.4 ± 3.4	67.6 ± 4.0	10.4
SR-GNN **w.o.** Reg.	72.0 ± 3.2	69.5 ± 3.7	8.8	66.1 ± 0.9	63.4 ± 0.9	4.2	66.4 ± 4.0	64.0 ± 5.5	13.4
SR-GNN (Ours)	$\mathbf{73.5 \pm 3.3}$	$\mathbf{71.4 \pm 3.5}$	**7.3**	$\mathbf{67.1 \pm 0.9}$	$\mathbf{64.0 \pm 0.9}$	**3.2**	$\mathbf{71.3 \pm 2.2}$	$\mathbf{70.2 \pm 2.4}$	**8.5**

Experiments

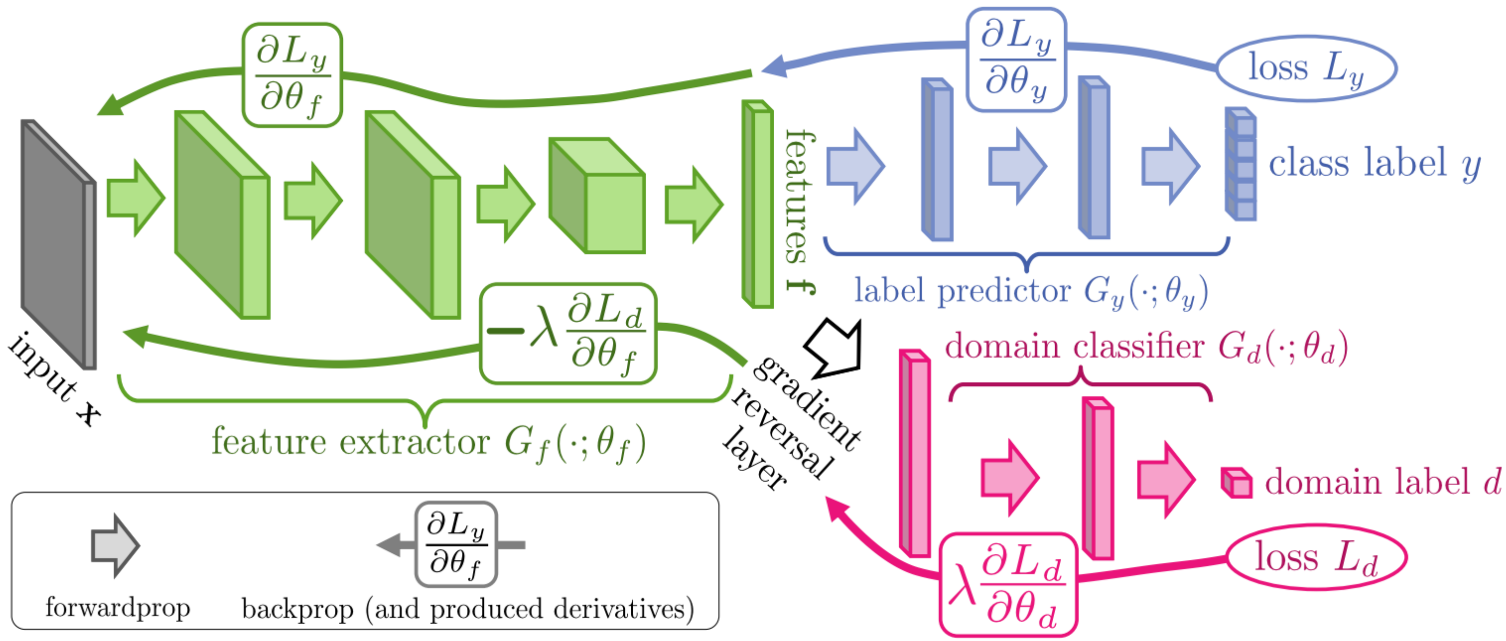
- Experimental results on large benchmarks

Table 2: Semi-supervised classification on ogb-arxiv and reddit varying label ratio.

label(%)	ogb-arxiv				reddit			
	1 %		5 %		1 %		5 %	
Method	Accuracy	$\Delta\downarrow$	Accuracy	$\Delta\downarrow$	Accuracy	$\Delta\downarrow$	Accuracy	$\Delta\downarrow$
GCN (IID)	66.0± 0.6	0	69.1± 0.6	0	93.8 ± 0.3	0	94.0 ± 0.1	0
Feat.+MLP	45.5± 0.6	21.5	43.7± 0.3	25.4	46.6±0.6	47.2	57.2±0.2	36.8
Emb.+MLP	51.1± 1.3	14.9	56.9± 0.8	13.2	89.6 ± 0.8	4.2	90.9 ± 0.3	3.1
DGI	44.8± 3.0	21.2	49.7± 3.3	19.4	83.7±1.2	10.1	85.4±0.6	8.6
GCN	59.3± 1.2	6.7	65.3 ± 0.6	3.8	89.7±1.0	4.1	90.9±0.3	3.1
GAT	58.6± 1.0	7.4	63.4 ± 1.0	5.7	80.5±5.4	13.3	82.0±3.6	12.0
SGC	59.0± 0.7	7.0	64.2 ± 1.3	4.9	88.6±1.0	5.2	90.6±0.2	3.4
APPNP	59.8± 1.1	6.2	65.1 ± 2.6	4.0	88.4±1.0	5.4	88.9±0.8	5.1
SR-GNN **w.o.** IR	60.6± 0.2	5.4	65.1±1.8	4.0	90.4± 0.6	3.4	91.2± 0.2	2.8
SR-GNN **w.o.** Reg.	61.0± 0.3	5.0	65.8±2.0	3.3	89.4± 0.8	4.4	91.9± 0.1	2.1
SR-GNN (Ours)	**61.6±0.6**	**4.4**	**66.5±0.6**	**2.6**	**91.5± 0.5**	**2.3**	**92.1± 0.3**	**1.9**

Experiments

- Comparison with other domain invariant learning methods (DANN)



Experiments

- Comparison with other domain invariant learning methods (DANN)

Table 4: Comparison of Domain-Adversarial Neural Network (DANN) and CMD regularizer used in SR-GNN with biased training data.

Method	Cora		Citeseer		PubMed	
	Micro-F1↑	Macro-F1↑	Micro-F1↑	Macro-F1↑	Micro-F1↑	Macro-F1↑
GCN	68.3	67.2	62.4	60.2	59.2	53.8
DANN	69.8	68.5	63.8	61.0	64.8	61.8
CMD (Ours)	**71.0**	**69.4**	**65.0**	**62.3**	**67.5**	**66.2**
APPNP	71.3	69.2	63.9	61.6	64.8	60.4
DANN	71.6	69.5	64.3	61.8	67.8	65.4
CMD (Ours)	**72.4**	**70.1**	**65.0**	**62.4**	**70.4**	**68.7**

[1] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., ... & Lempitsky, V. (2016). Domain-adversarial training of neural networks. *The journal of machine learning research*, *17*(1), 2096-2030.

Experiments

- Comparison with other domain invariant learning methods (DANN)

Table 3: Comparison of baseline and our SR(Shift-Robust) version ($\Delta(\%)$ -relative loss with biased sample) .

Method	Cora Micro-F1↑	Cora Macro-F1↑	Cora $\Delta(\%)$	Citeseer Micro-F1↑	Citeseer Macro-F1↑	Citeseer $\Delta(\%)$	PubMed Micro-F1↑	PubMed Macro-F1↑	PubMed $\Delta(\%)$
GCN (IID)	80.8	80.1	0%	70.3	66.8	0%	79.8	78.8	0%
GCN	67.6	66.4	-12%	62.7	60.4	-8%	60.6	56.0	-19%
SR-GCN	**69.6**	**68.2**	-10%	**64.7**	**62.0**	-6%	**67.0**	**65.2**	-13%
DGI (IID)	80.6	79.3	0%	70.8	66.7	0%	77.6	77.0	0%
DGI	71.7	69.2	-9%	62.6	60.0	-8%	58.0	52.4	-20%
SR-DGI	**74.3**	**72.6**	-6%	**65.8**	**62.6**	-6%	**62.0**	**57.8**	-16%

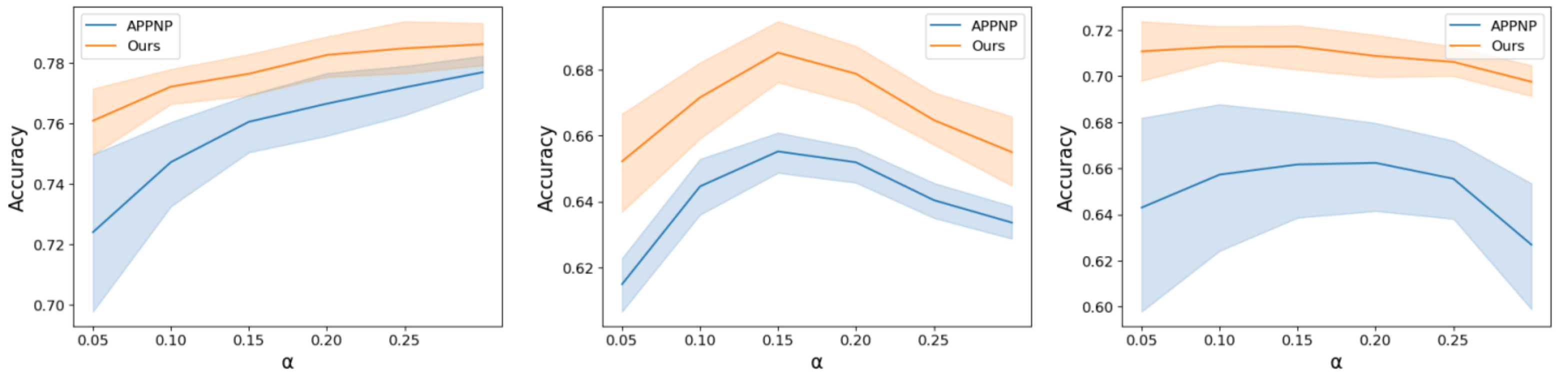
[1] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., ... & Lempitsky, V. (2016). Domain-adversarial training of neural networks. *The journal of machine learning research*, *17*(1), 2096-2030.

Outline

- Motivation
- Preliminaries
- Shift-Robust GNN
- Experiments
  - main result
  - Parameter sensitivity
- Future work

Experiments

- Performance with different α in PPR-S

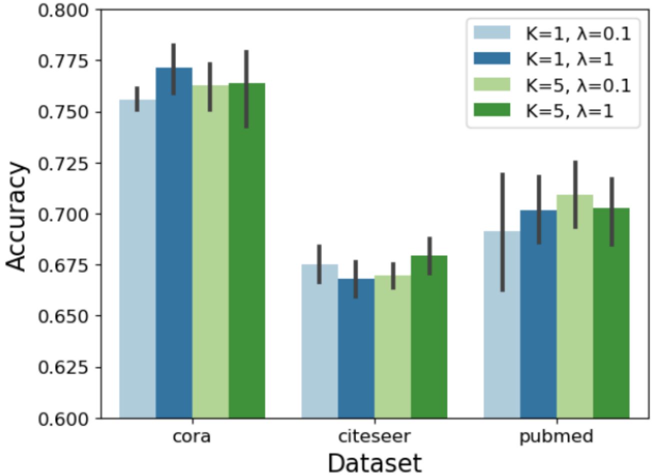


(a) Cora **(b)** Citeseer **(c)** Pubmed

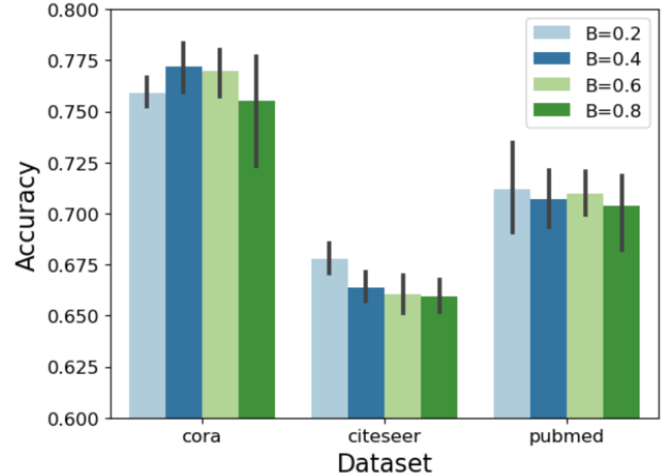
Figure 3: Varying α of biased sampler on three benchmarks.

Experiments

- Performance with different B_l, λ in SR-GNN



(a) Varying λ, k in regularization



(b) Varying B_l in instance weighting

Figure 4: Parameter sensitivity of SR-GNN.

$$\mathcal{L} = \frac{1}{M}\sum_{i=1}^{M} \beta_i l(y_i, \Phi(h_i)) + \lambda d(Z_{train}, Z_{IID})$$

Outline

- Motivation
- Preliminaries
- Shift-Robust GNN
- Experiments
- Future work

Future works

- Develop Shift-Robust GNNs on specific domains
  - Maximize the performance when dealing with specific shift in spam and abuse detection
- Theoretical guarantee towards shift-robust requirement
  - Generalization error in terms of distributional shift
  - Fairness of training data

$$P_{train}(X,Y,A) \neq P_{test}(X,Y,A)$$
$$\rightarrow \quad P_{train}(X,A) \neq P_{test}(X,A)$$

[1] Chen, Y., Raab, R., Wang, J., & Liu, Y. (2022). Fairness Transferability Subject to Bounded Distribution Shift. *arXiv preprint arXiv:2206.00129*.
[2] Rezaei, A., Liu, A., Memarrast, O., & Ziebart, B. D. (2021, May). Robust fairness under covariate shift. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 11, pp. 9419-9427).

References

[1] Liu, Z., Chen, C., Yang, X., Zhou, J., Li, X., & Song, L. (2018, October). Heterogeneous graph neural networks for malicious account detection. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (pp. 2077-2085).

[2] Halcrow, J., Mosoi, A., Ruth, S., & Perozzi, B. (2020, August). Grale: Designing networks for graph learning. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (pp. 2523-2532).

[3] Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

[4] Quinonero-Candela, J., Sugiyama, M., Schwaighofer, A., & Lawrence, N. D. (Eds.). (2008). *Dataset shift in machine learning*. Mit Press.

[5] Sugiyama, M., Krauledat, M., & Müller, K. R. (2007). Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, *8*(5).

[6] Long, M., Cao, Y., Wang, J., & Jordan, M. (2015, June). Learning transferable features with deep adaptation networks. In *International conference on machine learning* (pp. 97-105). PMLR.

References

[7] Zellinger, W., Moser, B. A., Grubinger, T., Lughofer, E., Natschläger, T., & Saminger-Platz, S. (2019). Robust unsupervised domain adaptation for neural networks via moment alignment. *Information Sciences*, *483*, 174-191.

[8] Wu, F., Souza, A., Zhang, T., Fifty, C., Yu, T., & Weinberger, K. (2019, May). Simplifying graph convolutional networks. In *International conference on machine learning* (pp. 6861-6871). PMLR.

[9] Klicpera, J., Bojchevski, A., & Günnemann, S. (2018). Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997*.

[10] Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., & Schölkopf, B. (2009). Covariate shift by kernel mean matching. *Dataset shift in machine learning*, *3*(4), 5.

[11] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., ... & Lempitsky, V. (2016). Domain-adversarial training of neural networks. *The journal of machine learning research*, *17*(1), 2096-2030.

[12] Andersen, R., Chung, F., & Lang, K. (2006, October). Local graph partitioning using pagerank vectors. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)* (pp. 475-486). IEEE.

Thank you for listening!
Q&A

Presented by Yian Wang

10/27