NEURAL INFORMATION PROCESSING SYSTEMS

Training Robust Graph Neural Networks by Modeling Noise Dependencies

Yeonjun In, Kanghoon Yoon, Sukwon Yun, Kibum Kim,
Sungchul Kim, Chanyoung Park

Korea Advanced Institute of Science and Technology (KAIST)
University of North Carolina at Chapel Hill
Adobe Research

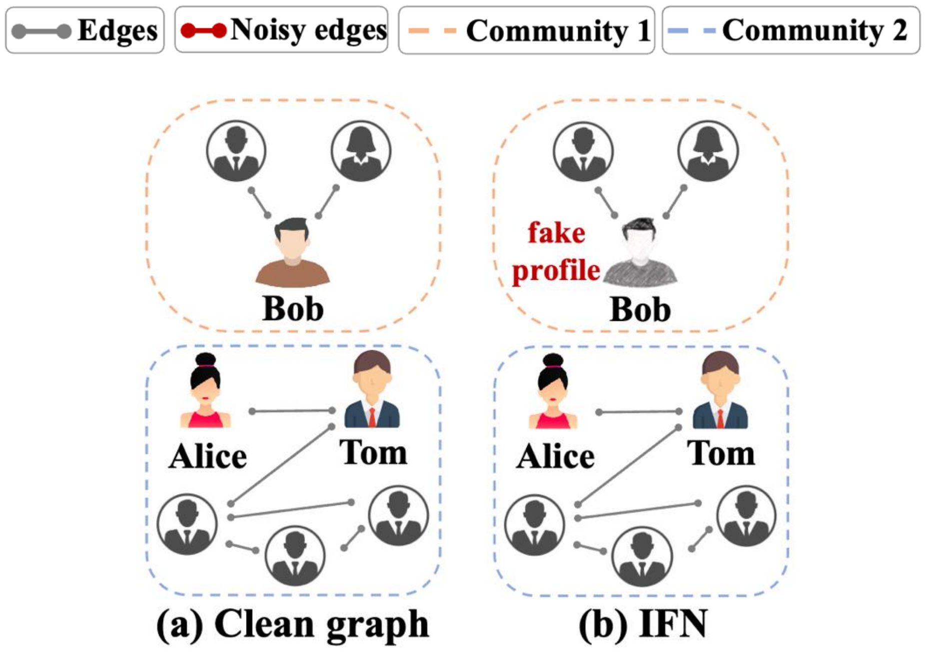
KAIST

THE UNIVERSITY *of* NORTH CAROLINA *at* CHAPEL HILL

Adobe

Inherent Noise on Graphs

In the majority of real-world scenarios, **node features frequently exhibit noise** due to various factors.



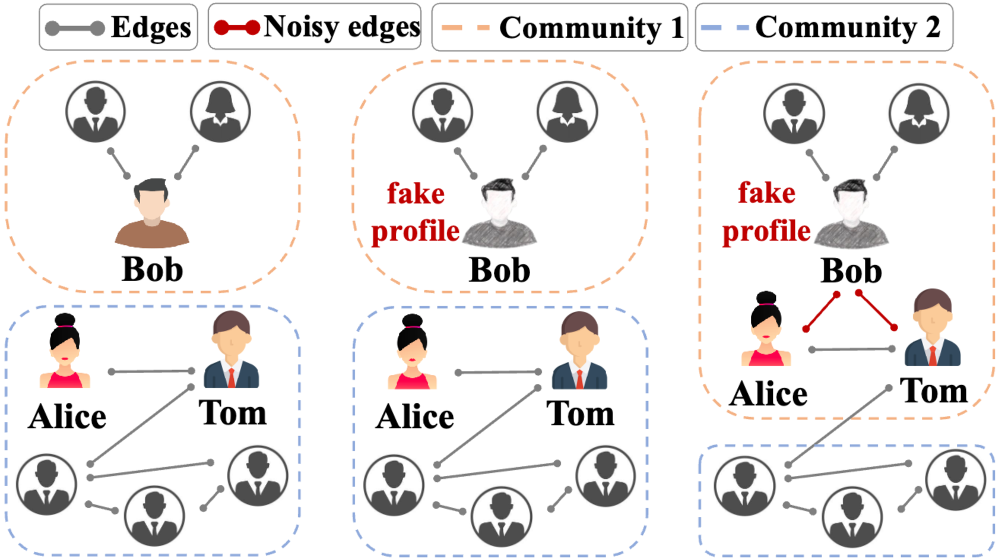
In social networks,

- users may create fake profiles or posting
- resulting in noisy node features.

Various feature-noise robust methods have been proposed as a promising solution.

Rethinking the Independent Node Feature Noise

Can noise in node features truly be isolated from influencing the graph structure or node labels?



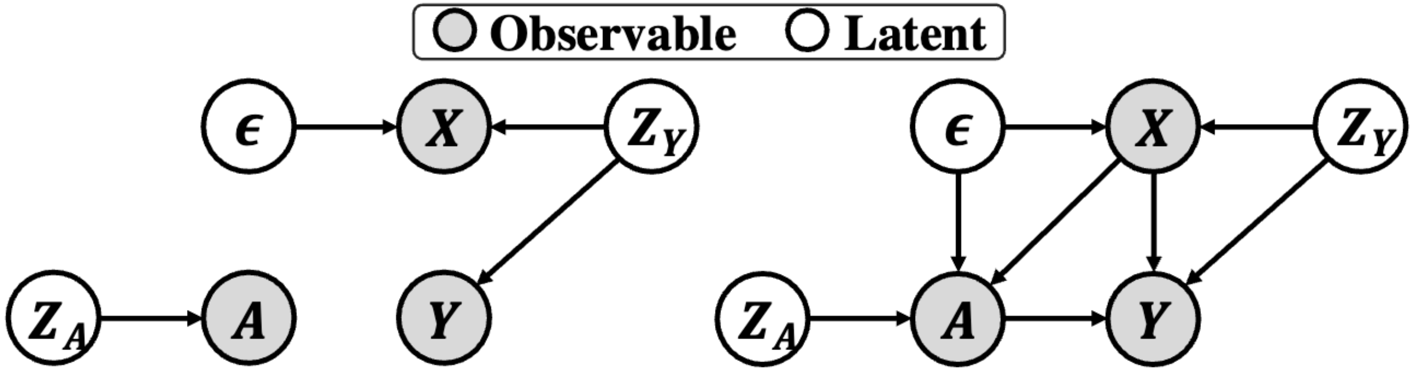
(a) Clean graph (b) IFN (c) DANG

Such scenario is observed in various domains

- social networks
- e-commerce system
- recommendation system
- biological network
- And so on

In real-world, noise in node features may create a chain of noise dependencies that propagate to the graph structure and node labels.

Dependency-Aware Noise on Graphs (DANG)



(a) DGP of IFN (b) DGP of DANG

- $\boldsymbol{X}$: obs. node features
- $\boldsymbol{A}$: obs. graph structure
- $\boldsymbol{Y}$: obs. node labels
- $\boldsymbol{\epsilon}$: noise-incurring variable
- $\boldsymbol{Z_A}$: latent clean graph structure
- $\boldsymbol{Z_Y}$: latent clean node labels

Causal relationships

- $\boldsymbol{X} \leftarrow (\boldsymbol{\epsilon}, \boldsymbol{Z_Y})$: X is built on Z_Y and is noisy due to ϵ
- $\boldsymbol{A} \leftarrow (\boldsymbol{Z_A}, \boldsymbol{X})$: A is built on Z_A and is feature-dependent noisy
- $\boldsymbol{Y} \leftarrow (\boldsymbol{Z_Y}, \boldsymbol{X}, \boldsymbol{A})$: Y is built on Z_Y and is noisy due to noisy X and A

Discussion on DANG

1) Under DANG, a graph does not contain any noise-free data sources.

- Challenging for the existing methods to tackle DANG, as they assume the completeness of at least one data source.

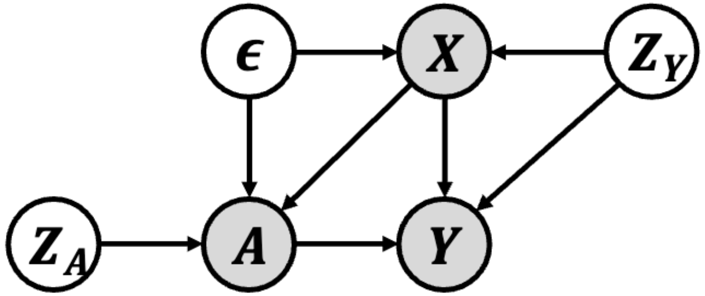
2) DANG is prevalent across various domains

- e.g., social networks, e-commerce, web graphs, citation networks, and biology

3) DANG addresses the practical gap between real-world and the simplistic noise assumptions of previous works.

Dependency-Aware GNN (DA-GNN)

Key idea: directly modeling the DGP of DANG by maximizing joint likelihood $p(X, A, Y)$

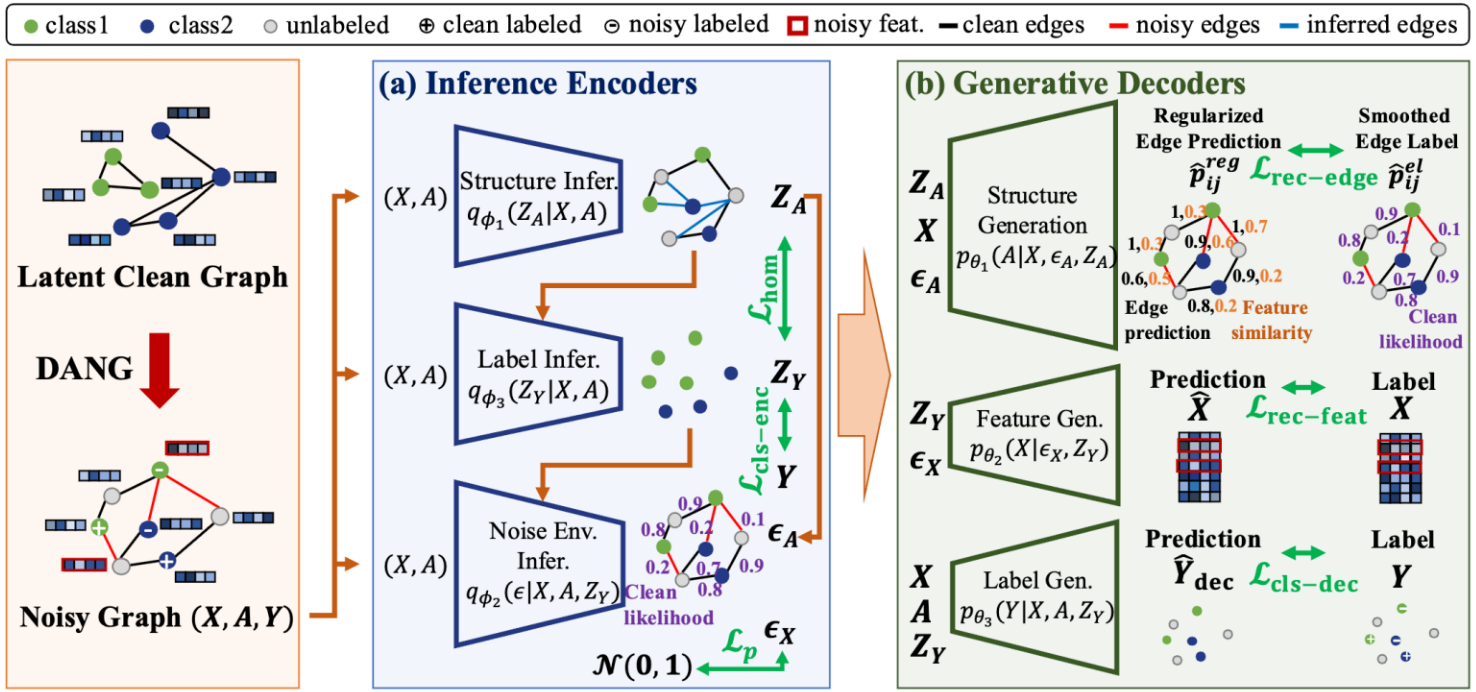


(b) DGP of DANG

Negative ELBO loss is derived with inference encoders and generative decoders

$$
\begin{aligned}
\mathcal{L}_{\text{ELBO}} = & -\mathbb{E}_{Z_A \sim \boxed{q_{\phi_1}(Z_A|X,A)}} \mathbb{E}_{\epsilon \sim q_{\phi_2}(\epsilon|X,A,Z_Y)} \left[\log\left(\boxed{p_{\theta_1}(A|X, \epsilon, Z_A)} \right) \right] \\
& - \mathbb{E}_{\epsilon \sim \boxed{q_{\phi_2}(\epsilon|X,A,Z_Y)}} \mathbb{E}_{Z_Y \sim q_{\phi_3}(Z_Y|X,A)} \left[\log\left(\boxed{p_{\theta_2}(X|\epsilon, Z_Y)} \right) \right] \\
& - \mathbb{E}_{Z_Y \sim \boxed{q_{\phi_3}(Z_Y|X,A)}} \left[\log\left(\boxed{p_{\theta_3}(Y|X, A, Z_Y)} \right) \right] \\
& + kl(q_{\phi_3}(Z_Y|X, A)||p(Z_Y)) + kl(q_{\phi_1}(Z_A|X, A)||p(Z_A)) \\
& + \mathbb{E}_{Z_Y \sim q_{\phi_3}(Z_Y|X,A)} \left[kl(q_{\phi_2}(\epsilon|X, A, Z_Y)||p(\epsilon)) \right]. \qquad (4)
\end{aligned}
$$

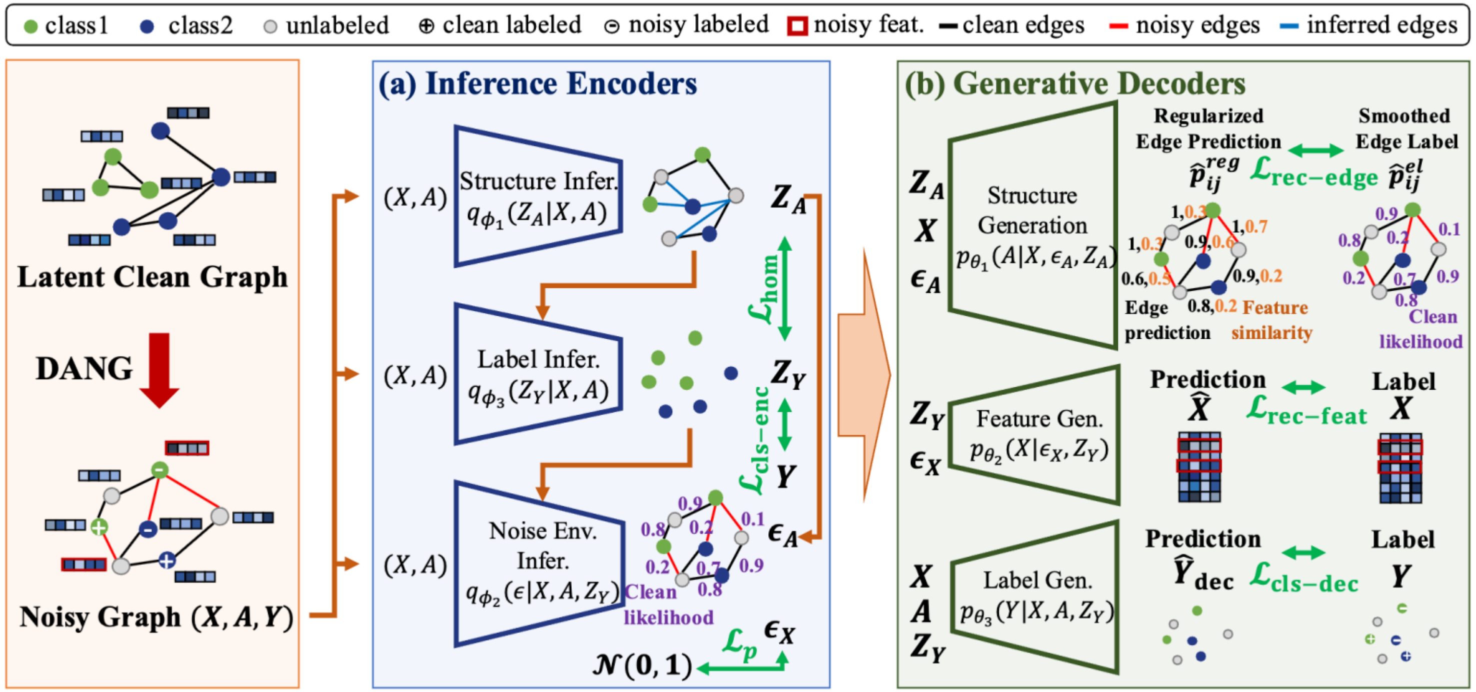
Dependency-Aware GNN (DA-GNN)



Inference Encoders

- $q_{\phi_1}(Z_A|X,A)$: infer latent graph structure → **Graph structure learner**
- $q_{\phi_3}(Z_Y|X,A)$: infer latent node label → **GNN classifier + regularizer to satisfy homophily**
- $q_{\phi_{22}}(\epsilon_A|X,A)$: likelihood of an observed edge being clean → **Early-phase edge prediction confidence**
- $q_{\phi_{21}}(\epsilon_X|X,Z_Y)$: infer latent ϵ_X → **Similar to VAE**

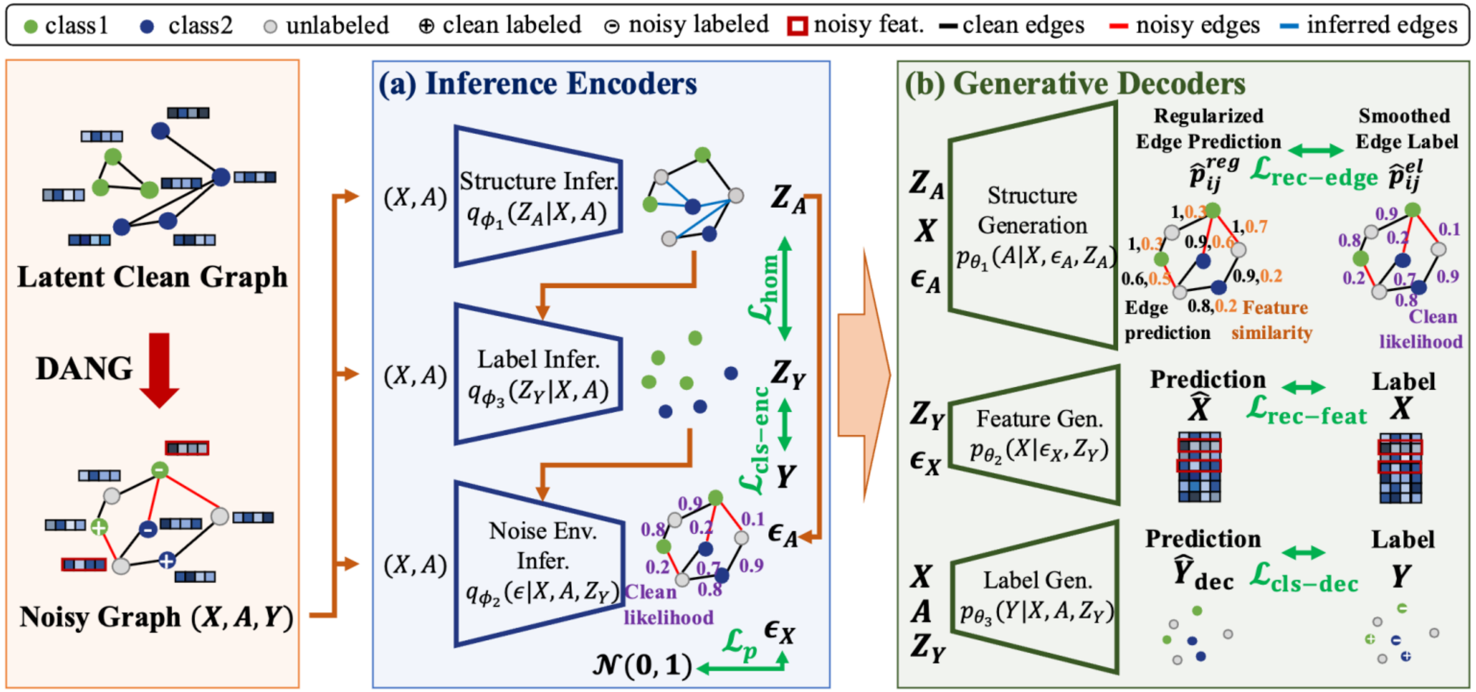
Dependency-Aware GNN (DA-GNN)



Generative Decoders

- $\max p_{\theta_1}(A|X, \epsilon_A, Z_A)$ → learn how the edge noise was generated
  → **Minimize edge pred loss with regularized prediction & label smoothing**
- $\max p_{\theta_3}(Y|X, A, Z_Y)$ → learn how the label noise was generated
  → **Minimize node classification loss with GNN classifier**
- $\max p_{\theta_2}(X|\epsilon_X, Z_Y)$ → learn how feature noise was generated **→ feat. recon. loss**

Dependency-Aware GNN (DA-GNN)



The inferred Z_Y and Z_A serve as a node classification and edge prediction.

DA-GNN is Robust under DANG

Node classification under synthetic DANG

Dataset	Setting	WSGNN	GraphGLOW	AirGNN	ProGNN	RSGNN	STABLE	EvenNet	NRGNN	RTGNN	SG-GSR	DA-GNN
Cora	Clean	**86.2±0.1**	85.2±0.7	85.0±0.2	85.3±0.4	**86.2±0.5**	86.1±0.2	**86.2±0.0**	**86.2±0.2**	86.1±0.2	85.7±0.1	**86.2±0.7**
	DANG-10%	80.7±0.3	79.7±0.2	79.7±0.5	79.6±0.7	81.9±0.3	82.2±0.7	80.7±0.7	81.0±0.5	81.8±0.3	82.7±0.1	**82.9±0.6**
	DANG-30%	70.0±0.6	71.6±0.5	71.5±0.8	74.5±0.1	71.9±0.5	74.3±0.3	65.2±1.7	73.5±0.8	72.6±1.5	76.1±0.2	**78.2±0.3**
	DANG-50%	55.9±1.1	59.6±0.1	56.2±0.8	66.4±0.4	59.9±0.5	62.8±2.4	47.1±1.8	61.9±1.4	60.9±0.4	64.3±0.5	**69.7±0.6**
Citeseer	Clean	76.6±0.6	76.5±1.0	71.5±0.2	72.6±0.5	75.8±0.4	74.6±0.6	76.4±0.5	75.0±1.3	76.1±0.4	75.3±0.3	**77.3±0.6**
	DANG-10%	72.8±0.8	71.4±0.8	66.2±0.7	67.5±0.6	73.3±0.5	71.5±0.3	71.1±0.4	71.9±0.3	73.2±0.2	74.2±0.5	**74.3±0.9**
	DANG-30%	63.3±0.7	60.6±0.2	58.0±0.4	61.0±0.2	63.9±0.5	62.5±1.4	61.2±0.6	62.5±0.7	64.2±1.9	**65.6±1.0**	**65.6±0.6**
	DANG-50%	53.4±0.6	48.8±0.6	50.0±0.6	53.3±0.2	55.3±0.4	54.7±1.7	47.2±1.1	52.6±0.9	54.2±1.8	54.8±1.8	**59.0±1.8**
Photo	Clean	92.9±0.3	94.2±0.4	93.5±0.1	90.1±0.2	93.6±0.8	93.4±0.1	94.5±0.4	90.3±1.7	91.3±0.6	94.3±0.1	**94.8±0.3**
	DANG-10%	83.9±1.8	92.1±0.8	87.3±0.9	84.3±0.1	92.1±0.2	92.2±0.1	92.6±0.0	84.3±1.3	89.4±0.5	93.0±0.1	**93.2±0.2**
	DANG-30%	51.9±6.8	88.4±0.2	67.8±4.3	74.7±0.2	86.6±1.0	88.0±1.0	89.6±0.2	69.0±2.2	86.4±0.5	89.3±0.3	**90.5±0.4**
	DANG-50%	31.9±5.6	85.4±0.6	57.8±0.7	48.9±0.5	75.6±2.6	80.2±1.8	84.6±0.4	57.5±1.8	79.2±0.3	84.1±0.4	**87.6±0.2**
Comp	Clean	83.1±3.1	91.3±0.9	83.4±1.2	83.9±0.8	91.1±0.1	90.2±0.2	90.1±0.2	87.5±1.0	87.3±1.0	91.3±0.7	**92.2±0.0**
	DANG-10%	75.0±1.2	88.0±0.7	76.8±1.8	72.0±0.2	88.1±0.7	85.9±0.5	87.6±0.7	85.7±0.9	85.9±0.1	89.5±0.5	**89.8±0.2**
	DANG-30%	48.5±5.8	84.9±0.4	59.2±0.9	66.9±0.8	81.7±0.2	80.4±1.0	84.8±0.5	74.8±3.5	77.0±1.5	84.5±0.4	**86.9±0.3**
	DANG-50%	39.6±4.0	80.1±0.5	44.1±1.4	43.3±0.3	73.9±2.3	68.8±1.3	77.5±1.9	65.3±3.2	69.4±0.3	78.6±0.6	**82.2±0.4**
Arxiv	Clean	OOM	OOM	58.0±0.4	OOM	OOM	OOM	65.7±0.6	OOM	60.4±0.5	OOM	**67.4±0.4**
	DANG-10%	OOM	OOM	50.6±0.5	OOM	OOM	OOM	58.4±1.2	OOM	54.3±0.4	OOM	**59.7±0.8**
	DANG-30%	OOM	OOM	36.8±0.3	OOM	OOM	OOM	47.4±2.5	OOM	45.0±0.6	OOM	**49.9±0.5**
	DANG-50%	OOM	OOM	26.1±0.2	OOM	OOM	OOM	38.0±4.1	OOM	38.4±0.8	OOM	**44.0±1.2**

DA-GNN is robust under synthetic DANG

A new benchmark dataset: real-world DANG in e-commerce

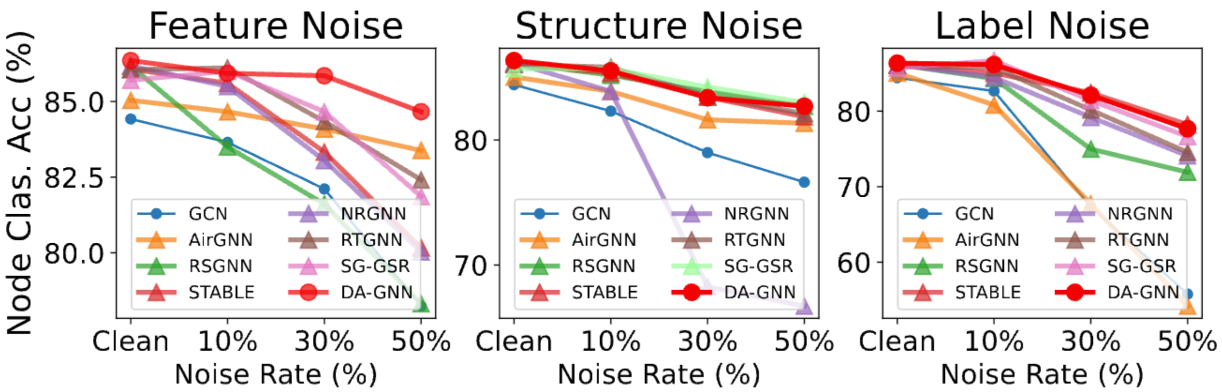
Task	Dataset	Setting	WSGNN	GraphGLOW	AirGNN	ProGNN	RSGNN	STABLE	EvenNet	NRGNN	RTGNN	SG-GSR	DA-GNN
NC	Auto	Clean	71.8±4.3	77.9±1.2	69.5±0.8	63.2±0.2	69.5±0.4	71.6±0.9	73.4±0.5	74.3±0.8	76.4±0.2	78.3±0.3	**79.3±0.2**
		+ DANG	57.7±1.3	59.4±0.8	53.9±0.1	48.6±0.3	56.8±0.9	57.5±0.2	57.1±2.1	55.8±1.0	59.6±0.8	**62.0±1.1**	61.4±0.4
	Garden	Clean	87.4±0.2	88.5±0.9	78.3±1.5	78.7±0.1	83.3±1.2	84.2±0.5	85.7±0.5	87.7±0.4	87.8±0.2	88.1±0.3	**88.7±0.3**
		+ DANG	77.6±0.8	78.1±1.5	66.1±1.7	73.0±0.4	76.2±0.5	77.2±3.3	75.6±2.4	76.1±0.2	76.0±0.6	**80.2±0.4**	**80.2±0.8**
LP	Auto	Clean	81.8±0.1	86.2±0.3	60.2±0.2	74.8±0.3	87.2±0.8	78.6±0.1	86.8±0.1	76.6±1.3	84.4±0.1	82.2±8.3	**88.2±0.3**
		+ DANG	69.1±0.6	**74.8±0.2**	57.9±0.4	56.7±0.5	65.0±0.2	57.3±0.1	70.5±0.2	47.5±1.7	72.2±0.2	65.6±7.4	73.6±0.6
	Garden	Clean	84.7±0.2	90.2±0.5	62.0±0.1	83.5±0.6	91.2±0.4	85.2±0.2	89.2±0.3	87.0±0.9	90.4±0.3	89.2±3.8	**92.6±0.2**
		+ DANG	84.6±0.7	90.1±0.4	58.2±0.5	83.3±0.5	91.2±0.5	85.0±0.1	90.0±0.7	58.6±4.5	90.4±0.2	86.0±7.2	**92.4±0.4**

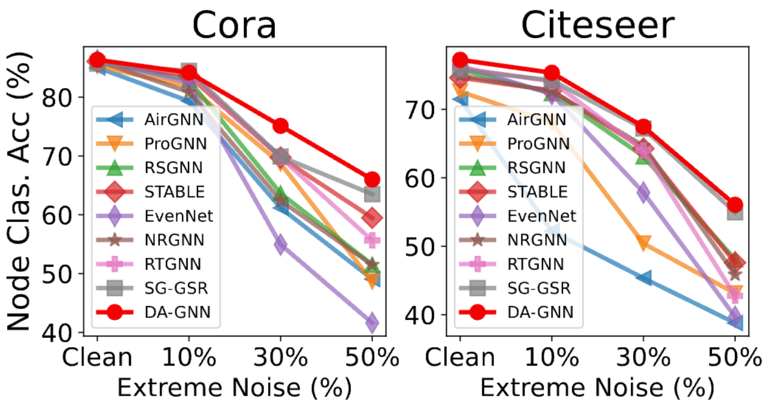
We newly design a benchmark dataset

- Mimicking a real-world DANG in e-commerce system
- Fraudsters' fake reviews results in
  - node feature noise
  - graph structure noise
  - node label noise

DA-GNN is also robust under real-world DANG

DA-GNN Also Robust under Existing Noise Scenarios





DA-GNN is also robust under existing noise scenarios.

Demonstrating the applicability of DA-GNN to various real-world scenario besides DANG.

Thank you for listening!

Please refer to our paper

"Training Robust Graph Neural Networks by Modeling Noise Dependencies"

Contact: yeonjun.in@kaist.ac.kr
Personal homepage: yeonjun-in.notion.site