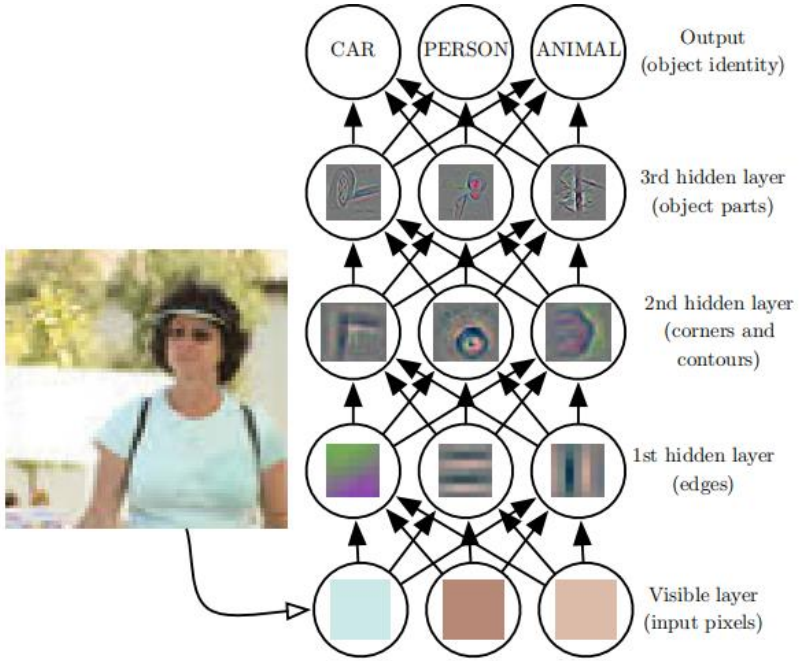
Classic GNNs are Strong Baselines: Reassessing GNNs for Node Classification

Yuankai Luo
PhD student at Beihang University
Joint PhD student at The Hong Kong Polytechnic University

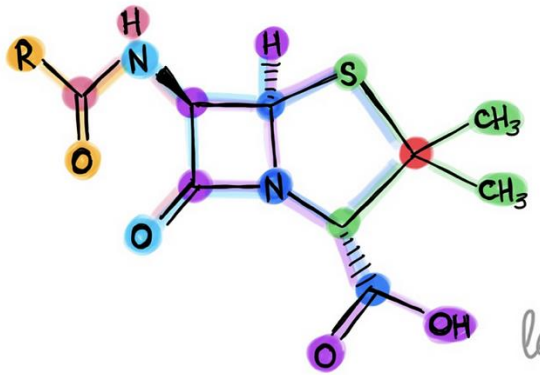
NeurIPS 2024

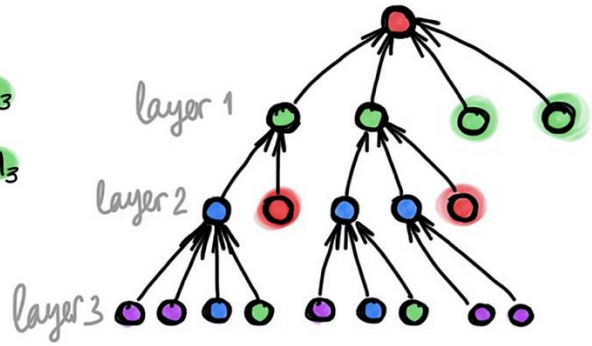
Background

- Deep learning has made significant breakthroughs in fields like computer vision, and natural language processing.



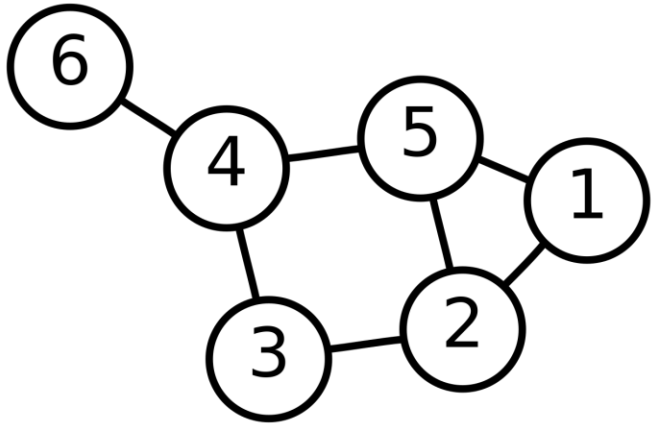
- It has potential in non-Euclidean graph data, like knowledge graphs, social networks, and biological molecules.



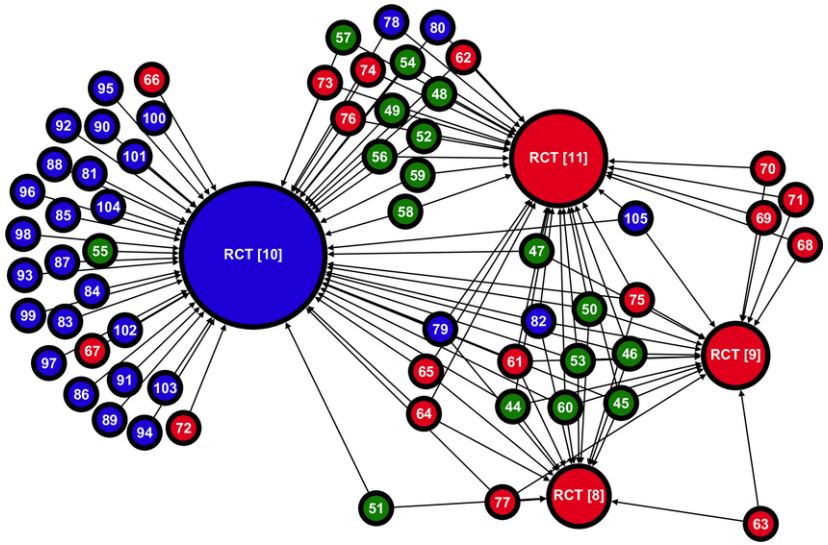


Background

- Graph Data:
  Graph data consists of nodes and edges connecting them, which can model various real-world systems.



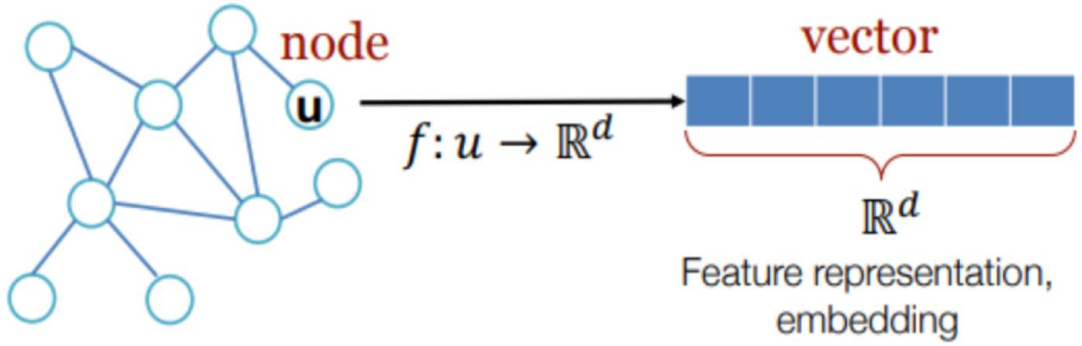
Formally, we can define a graph as $G = (V, E, X)$, where V denotes the set of nodes, $E \subseteq V \times V$ represents the set of edges, $X \in R^{|V| \times d}$ is the node feature matrix.



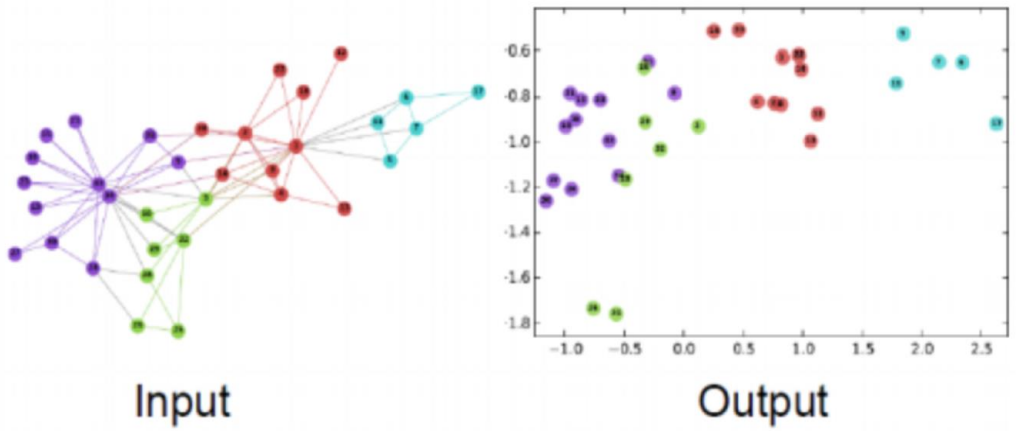
Background

- Graph Representation Learning:

It extracts representations in a low-dimensional, continuous vector space from raw graph data that machines can process effectively.



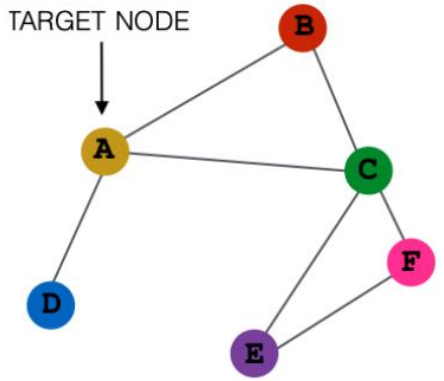
These representations keep the node's neighbor and attribute information, allowing standard machine learning techniques to tackle graph tasks like node classification.



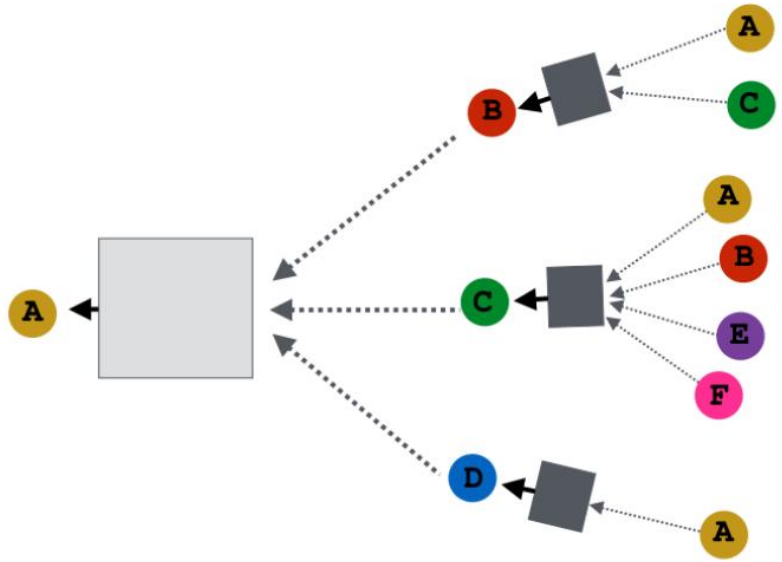
Graph Neural Networks (GNNs)

- GNNs as Graph Representation Learning Methods:

A message-passing mechanism creates new node representations, where each node gathers information from its neighbors and combines it to update its own embedding.



INPUT GRAPH



3 Classic GNNs on Node Classification

GCN, 2016

GraphSAGE, 2017

GAT, 2017

Graph Convolutional Networks (GCN)

The standard GCN model, is formualated as:

$$\boldsymbol{h}_v^l = \sigma\Big(\sum_{u \in \mathcal{N}(v) \cup \{v\}} \frac{1}{\sqrt{\hat{d}_u \hat{d}_v}} \boldsymbol{h}_u^{l-1} \boldsymbol{W}^l \Big)$$

where $\hat{d}_v$ denotes the degree of node v, W^l is the trainable weight matrix in layer l, and σ is the activation function.

The output of the last layer L is the representation of v produced by the GCN.

GraphSAGE

GraphSAGE learns representations through a different approach:

$$\boldsymbol{h}_v^l = \sigma(\boldsymbol{h}_v^{l-1}\boldsymbol{W}_1^l + (\text{mean}_{u\in\mathcal{N}(v)}\boldsymbol{h}_u^{l-1})\boldsymbol{W}_2^l)$$

where W_1^l and W_2^l are trainable weight matrices, and $\text{mean}_{u\in N(v)}h_u^{l-1}$ computes the average embedding of the neighboring nodes of v.

Graph Attention Networks (GAT)

GAT employ masked self-attention to assign weights to different neighboring nodes. For an edge $(v, u) \in E$, the propagation rule of GAT is defined as:

$$\alpha_{vu}^{l} = \frac{\exp\left(\text{LeakyReLU}\left(\mathbf{a}_l^{\top}\left[\boldsymbol{W}^l \boldsymbol{h}_v^{l-1} \| \boldsymbol{W}^l \boldsymbol{h}_u^{l-1}\right]\right)\right)}{\sum_{r \in \mathcal{N}(v)} \exp\left(\text{LeakyReLU}\left(\mathbf{a}_l^{\top}\left[\boldsymbol{W}^l \boldsymbol{h}_v^{l-1} \| \boldsymbol{W}^l \boldsymbol{h}_r^{l-1}\right]\right)\right)},$$

$$\boldsymbol{h}_v^{l} = \sigma\left(\sum_{u \in \mathcal{N}(v)} \alpha_{vu}^{l} \boldsymbol{h}_u^{l-1} \boldsymbol{W}^{l}\right),$$

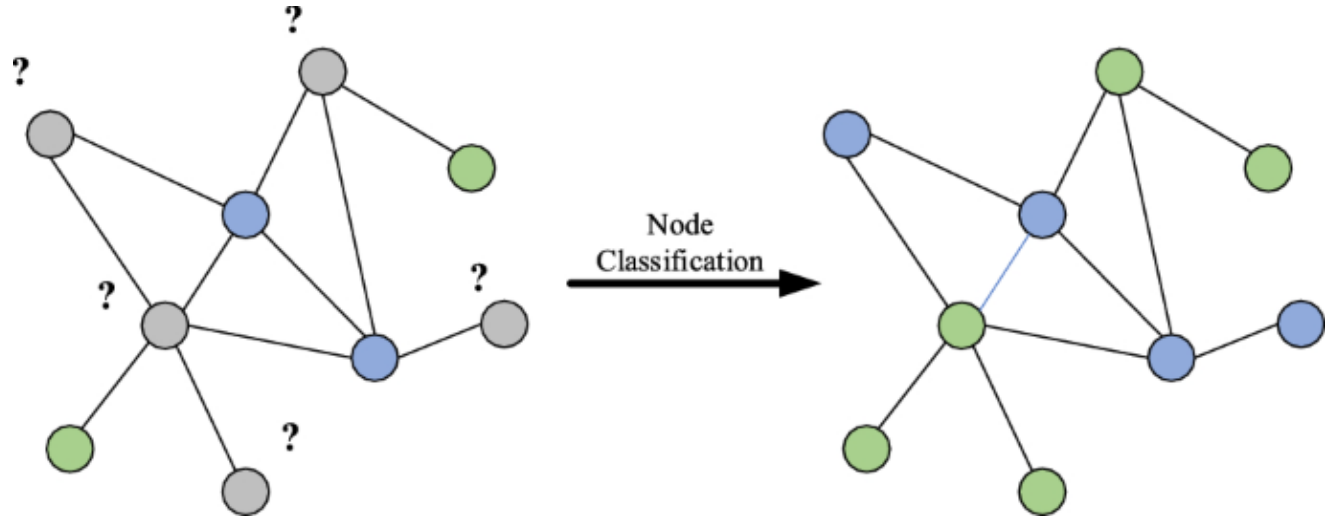
where a_l is a trainable weight vector and W^l is a trainable weight matrix.

Node Classification

Node Classification aims to predict the labels of the unlabeled nodes.

Typically, for any node v, the node representation generated by the last GNN layer is passed through a prediction head $g(\cdot)$, to obtain the predicted label.

The objective is to minimize the loss between the predicted label and true label.



Homophilous and Heterophilous Graphs

Homophilous graphs: connected nodes may belong to same classes.

Heterophilous graphs: connected nodes may belong to different classes.

Due to the homophily assumption (information from neighborhoods), GNNs cannot generalize well to heterophilous graphs. However, our study find that GNNs also work well on heterophilous graphs!

Key Hyperparameters for Training GNNs

- Normalization:

Layer Normalization (LN) or Batch Normalization (BN) can be used in every layer before the activation function.Taking GCN as an example:

$$\boldsymbol{h}_v^l = \sigma(\text{Norm}(\sum_{u \in \mathcal{N}(v) \cup \{v\}} \frac{1}{\sqrt{\hat{d}_u \hat{d}_v}} \boldsymbol{h}_u^{l-1} \boldsymbol{W}^l))$$

The normalization techniques are essential!

Key Hyperparameters for Training GNNs

- Dropout:

Dropout is a technique widely used in CNNs to address overfitting. Typically, dropout is applied to the feature embeddings after the activation function

$$\boldsymbol{h}_v^l = \text{Dropout}(\sigma(\text{Norm}(\sum_{u \in \mathcal{N}(v) \cup \{v\}} \frac{1}{\sqrt{\hat{d}_u \hat{d}_v}} \boldsymbol{h}_u^{l-1} \boldsymbol{W}^l)))$$

The dropout are essential!

Key Hyperparameters for Training GNNs

- **Residual Connections:**

Residual Connections significantly enhance CNNs performance by connecting layer inputs directly to outputs, thereby alleviating the vanishing gradient issue.

Formally, linear residual connections can be integrated into GNNs as follows:

$$\boldsymbol{h}_v^l = \text{Dropout}(\sigma(\text{Norm}(\boldsymbol{h}_v^{l-1}\boldsymbol{W}_r{}^l + \sum_{u \in \mathcal{N}(v) \cup \{v\}} \frac{1}{\sqrt{\hat{d}_u \hat{d}_v}} \boldsymbol{h}_u^{l-1} \boldsymbol{W}^l)))$$

Key Hyperparameters for Training GNNs

- **Network Depth:**

GNNs face unique challenges with depth, such as over-smoothing, where node representations become indistinguishable with increased network depth.

In practice, most GNNs adopt a shallow architecture, typically consisting of 2 to 5 layers.

Our findings indicate that comparable performance can be achieved from 2 to 10 layers.

Node Classification Datasets

Table 1: Overview of the datasets used for node classification.

Dataset	Type	# nodes	# edges	# Features	Classes	Metric
Cora	Homophily	2,708	5,278	1,433	7	Accuracy
CiteSeer	Homophily	3,327	4,522	3,703	6	Accuracy
PubMed	Homophily	19,717	44,324	500	3	Accuracy
Computer	Homophily	13,752	245,861	767	10	Accuracy
Photo	Homophily	7,650	119,081	745	8	Accuracy
CS	Homophily	18,333	81,894	6,805	15	Accuracy
Physics	Homophily	34,493	247,962	8,415	5	Accuracy
WikiCS	Homophily	11,701	216,123	300	10	Accuracy
Squirrel	Heterophily	2,223	46,998	2,089	5	Accuracy
Chameleon	Heterophily	890	8,854	2,325	5	Accuracy
Roman-Empire	Heterophily	22,662	32,927	300	18	Accuracy
Amazon-Ratings	Heterophily	24,492	93,050	300	5	Accuracy
Minesweeper	Heterophily	10,000	39,402	7	2	ROC-AUC
Questions	Heterophily	48,921	153,540	301	2	ROC-AUC
ogbn-proteins	Homophily (Large graphs)	132,534	39,561,252	8	2	ROC-AUC
ogbn-arxiv	Homophily (Large graphs)	169,343	1,166,243	128	40	Accuracy
ogbn-products	Homophily (Large graphs)	2,449,029	61,859,140	100	47	Accuracy
pokec	Heterophily (Large graphs)	1,632,803	30,622,564	65	2	Accuracy

Empirical Findings: Homophilous Graphs

Table 2: Node classification results over homophilous graphs (%). * indicates our implementation, while other results are taken from [12, 71]. The top 1st, 2nd and 3rd results are highlighted.

	Cora	CiteSeer	PubMed	Computer	Photo	CS	Physics	WikiCS
# nodes	2,708	3,327	19,717	13,752	7,650	18,333	34,493	11,701
# edges	5,278	4,732	44,324	245,861	119,081	81,894	247,962	216,123
Metric	Accuracy↑	Accuracy↑	Accuracy↑	Accuracy↑	Accuracy↑	Accuracy↑	Accuracy↑	Accuracy↑
GraphGPS	82.84 ± 1.03	72.73 ± 1.23	79.94 ± 0.26	91.19 ± 0.54	95.06 ± 0.13	93.93 ± 0.12	97.12 ± 0.19	78.66 ± 0.49
GraphGPS*	83.87 ± 0.96	72.73 ± 1.23	79.94 ± 0.26	91.79 ± 0.63	94.89 ± 0.14	94.04 ± 0.21	96.71 ± 0.15	78.66 ± 0.49
NAGphormer	82.12 ± 1.18	71.47 ± 1.30	79.73 ± 0.28	91.22 ± 0.14	95.49 ± 0.11	95.75 ± 0.09	97.34 ± 0.03	77.16 ± 0.72
NAGphormer*	80.92 ± 1.17	70.59 ± 0.89	80.14 ± 1.06	91.69 ± 0.30	96.14 ± 0.16	95.85 ± 0.16	97.35 ± 0.12	77.92 ± 0.93
Exphormer	82.77 ± 1.38	71.63 ± 1.19	79.46 ± 0.35	91.47 ± 0.17	95.35 ± 0.22	94.93 ± 0.01	96.89 ± 0.09	78.54 ± 0.49
Exphormer*	83.29 ± 1.36	71.85 ± 1.11	79.67 ± 0.73	91.80 ± 0.35	95.69 ± 0.39	95.92 ± 0.25	97.06 ± 0.13	79.38 ± 0.62
GOAT	83.18 ± 1.27	71.99 ± 1.26	79.13 ± 0.38	90.96 ± 0.90	92.96 ± 1.48	94.21 ± 0.38	96.24 ± 0.24	77.00 ± 0.77
GOAT*	83.26 ± 1.24	72.21 ± 1.29	80.06 ± 0.67	92.29 ± 0.37	94.33 ± 0.21	93.81 ± 0.19	96.47 ± 0.16	77.96 ± 0.63
NodeFormer	82.20 ± 0.90	72.50 ± 1.10	79.90 ± 1.00	86.98 ± 0.62	93.46 ± 0.35	95.64 ± 0.22	96.45 ± 0.28	74.73 ± 0.94
NodeFormer*	82.73 ± 0.75	72.37 ± 1.20	79.59 ± 0.92	87.29 ± 0.58	93.43 ± 0.56	95.69 ± 0.27	96.48 ± 0.34	75.13 ± 0.93
SGFormer	84.50 ± 0.80	72.60 ± 0.20	80.30 ± 0.60	91.99 ± 0.76	95.10 ± 0.47	94.78 ± 0.20	96.60 ± 0.18	73.46 ± 0.56
SGFormer*	84.82 ± 0.85	72.72 ± 1.15	80.60 ± 0.49	92.42 ± 0.66	95.58 ± 0.36	95.71 ± 0.24	96.75 ± 0.26	80.05 ± 0.46
Polynormer	83.25 ± 0.93	72.31 ± 0.78	79.24 ± 0.43	93.68 ± 0.21	96.46 ± 0.26	95.53 ± 0.16	97.27 ± 0.08	80.10 ± 0.67
Polynormer*	83.43 ± 0.89	72.19 ± 0.83	79.35 ± 0.73	93.78 ± 0.10	96.57 ± 0.23	95.42 ± 0.19	97.18 ± 0.11	80.26 ± 0.92
GCN	81.60 ± 0.40	71.60 ± 0.40	78.80 ± 0.60	89.65 ± 0.52	92.70 ± 0.20	92.92 ± 0.12	96.18 ± 0.07	77.47 ± 0.85
GCN*	85.10 ± 0.67 **3.50↑**	73.14 ± 0.67 **1.54↑**	81.12 ± 0.52 **2.32↑**	93.99 ± 0.12 **4.34↑**	96.10 ± 0.46 **3.40↑**	96.17 ± 0.06 **3.25↑**	97.46 ± 0.10 **1.28↑**	80.30 ± 0.62 **2.83↑**
GraphSAGE	82.68 ± 0.47	71.93 ± 0.85	79.41 ± 0.53	91.20 ± 0.29	94.59 ± 0.14	93.91 ± 0.13	96.49 ± 0.06	74.77 ± 0.95
GraphSAGE*	83.88 ± 0.65 **1.20↑**	72.26 ± 0.55 **0.33↑**	79.72 ± 0.50 **0.31↑**	93.25 ± 0.14 **2.05↑**	96.78 ± 0.23 **2.19↑**	96.38 ± 0.11 **2.47↑**	97.19 ± 0.05 **0.70↑**	80.69 ± 0.31 **5.92↑**
GAT	83.00 ± 0.70	72.10 ± 1.10	79.00 ± 0.40	90.78 ± 0.13	93.87 ± 0.11	93.61 ± 0.14	96.17 ± 0.08	76.91 ± 0.82
GAT*	84.46 ± 0.55 **1.46↑**	72.22 ± 0.84 **0.12↑**	80.28 ± 0.64 **1.28↑**	94.09 ± 0.37 **3.31↑**	96.60 ± 0.33 **2.73↑**	96.21 ± 0.14 **2.60↑**	97.25 ± 0.06 **1.08↑**	81.07 ± 0.54 **4.16↑**

Empirical Findings: Homophilous Graphs

- Observations on Homophilous Graphs. Classic GNNs, with only slight adjustments to hyperparameters, are highly competitive in node classification tasks on homophilous graphs, often outperforming state-of-the-art graph transformers in many cases.

s over homophilous graphs (%). * indicates our implementation, while other top 1st, 2nd and 3rd results are highlighted.

ubMed	Computer	Photo	CS	Physics	WikiCS
9,717 4,324 ccuracy↑	13,752 245,861 Accuracy↑	7,650 119,081 Accuracy↑	18,333 81,894 Accuracy↑	34,493 247,962 Accuracy↑	11,701 216,123 Accuracy↑
9.94 ± 0.26	91.19 ± 0.54	95.06 ± 0.13	93.93 ± 0.12	97.12 ± 0.19	78.66 ± 0.49
9.94 ± 0.26	91.79 ± 0.63	94.89 ± 0.14	94.04 ± 0.21	96.71 ± 0.15	78.66 ± 0.49
9.73 ± 0.28	91.22 ± 0.14	95.49 ± 0.11	95.75 ± 0.09	97.34 ± 0.03	77.16 ± 0.72
0.14 ± 1.06	91.69 ± 0.30	96.14 ± 0.16	95.85 ± 0.16	97.35 ± 0.12	77.92 ± 0.93
9.46 ± 0.35	91.47 ± 0.17	95.35 ± 0.22	94.93 ± 0.01	96.89 ± 0.09	78.54 ± 0.49
9.67 ± 0.73	91.80 ± 0.35	95.69 ± 0.39	95.92 ± 0.25	97.06 ± 0.13	79.38 ± 0.62
9.13 ± 0.38	90.96 ± 0.90	92.96 ± 1.48	94.21 ± 0.38	96.24 ± 0.24	77.00 ± 0.77
0.06 ± 0.67	92.29 ± 0.37	94.33 ± 0.21	93.81 ± 0.19	96.47 ± 0.16	77.96 ± 0.63
9.90 ± 1.00	86.98 ± 0.62	93.46 ± 0.35	95.64 ± 0.22	96.45 ± 0.28	74.73 ± 0.94
9.59 ± 0.92	87.29 ± 0.58	93.43 ± 0.56	95.69 ± 0.27	96.48 ± 0.34	75.13 ± 0.93
0.30 ± 0.60	91.99 ± 0.76	95.10 ± 0.47	94.78 ± 0.20	96.60 ± 0.18	73.46 ± 0.56
0.60 ± 0.49	92.42 ± 0.66	95.58 ± 0.36	95.71 ± 0.24	96.75 ± 0.26	80.05 ± 0.46
9.24 ± 0.43	93.68 ± 0.21	96.46 ± 0.26	95.53 ± 0.16	97.27 ± 0.08	80.10 ± 0.67
9.35 ± 0.73	93.78 ± 0.10	96.57 ± 0.23	95.42 ± 0.19	97.18 ± 0.11	80.26 ± 0.92
8.80 ± 0.60 1.12 ± 0.52 **2.32↑**	89.65 ± 0.52 93.99 ± 0.12 **4.34↑**	92.70 ± 0.20 96.10 ± 0.46 **3.40↑**	92.92 ± 0.12 96.17 ± 0.06 **3.25↑**	96.18 ± 0.07 97.46 ± 0.10 **1.28↑**	77.47 ± 0.85 80.30 ± 0.62 **2.83↑**
9.41 ± 0.53 9.72 ± 0.50 **0.31↑**	91.20 ± 0.29 93.25 ± 0.14 **2.05↑**	94.59 ± 0.14 96.78 ± 0.23 **2.19↑**	93.91 ± 0.13 96.38 ± 0.11 **2.47↑**	96.49 ± 0.06 97.19 ± 0.05 **0.70↑**	74.77 ± 0.95 80.69 ± 0.31 **5.92↑**
79.00 ± 0.40 80.28 ± 0.64 **1.28↑**	90.78 ± 0.13 94.09 ± 0.37 **3.31↑**	93.87 ± 0.11 96.60 ± 0.33 **2.73↑**	93.61 ± 0.14 96.21 ± 0.14 **2.60↑**	96.17 ± 0.08 97.25 ± 0.06 **1.08↑**	76.91 ± 0.82 81.07 ± 0.54 **4.16↑**

Empirical Findings: Heterophilous Graphs

Table 3: Node classification results on heterophilous graphs (%). * indicates our implementation, while other results are taken from [12, 71, 54]. The top 1st, 2nd and 3rd results are highlighted.

	Squirrel	Chameleon	Amazon-Ratings	Roman-Empire	Minesweeper	Questions
# nodes	2223	890	24,492	22,662	10,000	48,921
# edges	46,998	8,854	93,050	32,927	39,402	153,540
Metric	Accuracy↑	Accuracy↑	Accuracy↑	Accuracy↑	ROC-AUC↑	ROC-AUC↑
H2GCN	35.10 ± 1.15	26.75 ± 3.64	36.47 ± 0.23	60.11 ± 0.52	89.71 ± 0.31	63.59 ± 1.46
CPGNN	30.04 ± 2.03	33.00 ± 3.15	39.79 ± 0.77	63.96 ± 0.62	52.03 ± 5.46	65.96 ± 1.95
GPRGNN	38.95 ± 1.99	39.93 ± 3.30	44.88 ± 0.34	64.85 ± 0.27	86.24 ± 0.61	55.48 ± 0.91
FSGNN	35.92 ± 1.32	40.61 ± 2.97	52.74 ± 0.83	79.92 ± 0.56	90.08 ± 0.70	**78.86** ± 0.92
GloGNN	35.11 ± 1.24	25.90 ± 3.58	36.89 ± 0.14	59.63 ± 0.69	51.08 ± 1.23	65.74 ± 1.19
GraphGPS	39.67 ± 2.84	40.79 ± 4.03	53.10 ± 0.42	82.00 ± 0.61	90.63 ± 0.67	71.73 ± 1.47
GraphGPS*	39.81 ± 2.28	41.55 ± 3.91	53.27 ± 0.66	82.72 ± 0.68	90.75 ± 0.89	72.56 ± 1.33
NodeFormer	38.52 ± 1.57	34.73 ± 4.14	43.86 ± 0.35	64.49 ± 0.73	86.71 ± 0.88	74.27 ± 1.46
NodeFormer*	38.89 ± 2.67	36.38 ± 3.85	43.79 ± 0.57	74.83 ± 0.81	87.71 ± 0.69	75.02 ± 1.61
SGFormer	41.80 ± 2.27	44.93 ± 3.91	48.01 ± 0.49	79.10 ± 0.32	90.89 ± 0.58	72.15 ± 1.31
SGFormer*	**42.65** ± 2.41	**45.21** ± 3.72	54.14 ± 0.62	80.01 ± 0.44	91.42 ± 0.41	73.81 ± 0.59
Polynormer	40.87 ± 1.96	41.82 ± 3.45	54.81 ± 0.49	92.55 ± 0.37	97.46 ± 0.36	78.92 ± 0.89
Polynormer*	**41.97** ± 2.14	41.97 ± 3.18	**54.96** ± 0.22	**92.66** ± 0.60	97.49 ± 0.48	**78.94** ± 0.78
GCN	38.67 ± 1.84	41.31 ± 3.05	48.70 ± 0.63	73.69 ± 0.74	89.75 ± 0.52	76.09 ± 1.27
GCN*	**45.01** ± 1.63 **6.34↑**	**46.29** ± 3.40 **4.98↑**	53.80 ± 0.60 **5.10↑**	**91.27** ± 0.20 **17.58↑**	**97.86** ± 0.24 **8.11↑**	**79.02** ± 0.60 **2.93↑**
GraphSAGE	36.09 ± 1.99	37.77 ± 4.14	53.63 ± 0.39	85.74 ± 0.67	93.51 ± 0.57	76.44 ± 0.62
GraphSAGE*	40.78 ± 1.47 **4.69↑**	**44.81** ± 4.74 **7.04↑**	**55.40** ± 0.21 **1.77↑**	**91.06** ± 0.27 **5.32↑**	**97.77** ± 0.62 **4.26↑**	77.21 ± 1.28 **0.77↑**
GAT	35.62 ± 2.06	39.21 ± 3.08	52.70 ± 0.62	88.75 ± 0.41	93.91 ± 0.35	76.79 ± 0.71
GAT*	41.73 ± 2.07 **6.11↑**	44.13 ± 4.17 **4.92↑**	**55.54** ± 0.51 **2.84↑**	90.63 ± 0.14 **1.88↑**	**97.73** ± 0.73 **3.82↑**	77.95 ± 0.51 **1.16↑**

Empirical Findings: Heterophilous Graphs

- **Observations on Heterophilous Graphs.** Our implementation has significantly enhanced the previously reported best results of classic GNNs on heterophilous graphs, surpassing specialized GNN models tailored for such graphs and even outperforming the leading graph transformer architectures. This advancement challenging the prevailing assumption that GNNs are primarily suited for homophilous graph structures.

Table 3: Node classification results on heterophilous graphs (%). * indicates our implementation, while other [12, 71, 54]. The top 1st, 2nd and 3rd results are highlighted.

Chameleon	Amazon-Ratings	Roman-Empire	Minesweeper	Questions
890	24,492	22,662	10,000	48,921
8,854	93,050	32,927	39,402	153,540
Accuracy↑	Accuracy↑	Accuracy↑	ROC-AUC↑	ROC-AUC↑
26.75 ± 3.64	36.47 ± 0.23	60.11 ± 0.52	89.71 ± 0.31	63.59 ± 1.46
33.00 ± 3.15	39.79 ± 0.77	63.96 ± 0.62	52.03 ± 5.46	65.96 ± 1.95
39.93 ± 3.30	44.88 ± 0.34	64.85 ± 0.27	86.24 ± 0.61	55.48 ± 0.91
40.61 ± 2.97	52.74 ± 0.83	79.92 ± 0.56	90.08 ± 0.70	78.86 ± 0.92
25.90 ± 3.58	36.89 ± 0.14	59.63 ± 0.69	51.08 ± 1.23	65.74 ± 1.19
40.79 ± 4.03	53.10 ± 0.42	82.00 ± 0.61	90.63 ± 0.67	71.73 ± 1.47
41.55 ± 3.91	53.27 ± 0.66	82.72 ± 0.68	90.75 ± 0.89	72.56 ± 1.33
34.73 ± 4.14	43.86 ± 0.35	64.49 ± 0.73	86.71 ± 0.88	74.27 ± 1.46
36.38 ± 3.85	43.79 ± 0.57	74.83 ± 0.81	87.71 ± 0.69	75.02 ± 1.61
44.93 ± 3.91	48.01 ± 0.49	79.10 ± 0.32	90.89 ± 0.58	72.15 ± 1.31
45.21 ± 3.72	54.14 ± 0.62	80.01 ± 0.44	91.42 ± 0.41	73.81 ± 0.59
41.82 ± 3.45	54.81 ± 0.49	92.55 ± 0.37	97.46 ± 0.36	78.92 ± 0.89
41.97 ± 3.18	54.96 ± 0.22	92.66 ± 0.60	97.49 ± 0.48	78.94 ± 0.78
41.31 ± 3.05	48.70 ± 0.63	73.69 ± 0.74	89.75 ± 0.52	76.09 ± 1.27
46.29 ± 3.40 **4.98↑**	53.80 ± 0.60 **5.10↑**	91.27 ± 0.20 **17.58↑**	97.86 ± 0.24 **8.11↑**	79.02 ± 0.60 **2.93↑**
37.77 ± 4.14	53.63 ± 0.39	85.74 ± 0.67	93.51 ± 0.57	76.44 ± 0.62
44.81 ± 4.74 **7.04↑**	55.40 ± 0.21 **1.77↑**	91.06 ± 0.27 **5.32↑**	97.77 ± 0.62 **4.26↑**	77.21 ± 1.28 **0.77↑**
39.21 ± 3.08	52.70 ± 0.62	88.75 ± 0.41	93.91 ± 0.35	76.79 ± 0.71
44.13 ± 4.17 **4.92↑**	55.54 ± 0.51 **2.84↑**	90.63 ± 0.14 **1.88↑**	97.73 ± 0.73 **3.82↑**	77.95 ± 0.51 **1.16↑**

Empirical Findings: Large-scale Graphs

Table 4: Node classification results on large-scale graphs (%). * indicates our implementation, while other results are taken from [12, 71]. The top 1st, 2nd and 3rd results are highlighted. OOM means out of memory.

	ogbn-proteins	ogbn-arxiv	ogbn-products	pokec
# nodes	132,534	169,343	2,449,029	1,632,803
# edges	39,561,252	1,166,243	61,859,140	30,622,564
Metric	ROC-AUC↑	Accuracy↑	Accuracy↑	Accuracy↑
GraphGPS	76.83 ± 0.26	70.97 ± 0.41	OOM	OOM
GraphGPS*	77.15 ± 0.64	71.23 ± 0.59	OOM	OOM
NAGphormer	73.61 ± 0.33	70.13 ± 0.55	73.55 ± 0.21	76.59 ± 0.25
NAGphormer*	72.17 ± 0.45	70.88 ± 0.24	74.63 ± 0.29	75.92 ± 0.68
Exphormer	74.58 ± 0.26	72.44 ± 0.28	OOM	OOM
Exphormer*	77.62 ± 0.33	72.32 ± 0.36	OOM	OOM
GOAT	74.18 ± 0.37	72.41 ± 0.40	82.00 ± 0.43	66.37 ± 0.94
GOAT*	79.31 ± 0.42	72.76 ± 0.29	82.27 ± 0.56	72.64 ± 0.67
NodeFormer	77.45 ± 1.15	59.90 ± 0.42	72.93 ± 0.13	71.00 ± 1.30
NodeFormer*	77.86 ± 0.84	67.78 ± 0.28	73.96 ± 0.30	71.00 ± 1.30
SGFormer	79.53 ± 0.38	72.63 ± 0.13	74.16 ± 0.31	73.76 ± 0.24
SGFormer*	79.92 ± 0.48	72.76 ± 0.33	81.54 ± 0.43	82.44 ± 0.76
Polynormer	78.97 ± 0.47	73.46 ± 0.16	83.82 ± 0.11	86.10 ± 0.05
Polynormer*	79.53 ± 0.67	73.40 ± 0.22	83.82 ± 0.11	86.06 ± 0.25
GCN	72.51 ± 0.35	71.74 ± 0.29	75.64 ± 0.21	75.45 ± 0.17
GCN*	77.29 ± 0.46 **4.78↑**	73.53 ± 0.12 **1.79↑**	82.33 ± 0.19 **6.69↑**	86.33 ± 0.17 **10.88↑**
GraphSAGE	77.68 ± 0.20	71.49 ± 0.27	78.29 ± 0.16	75.63 ± 0.38
GraphSAGE*	82.21 ± 0.32 **4.53↑**	73.00 ± 0.28 **1.51↑**	83.89 ± 0.36 **5.60↑**	85.97 ± 0.21 **10.34↑**
GAT	72.02 ± 0.44	71.95 ± 0.36	79.45 ± 0.59	72.23 ± 0.18
GAT*	85.01 ± 0.46 **12.99↑**	73.30 ± 0.18 **1.35↑**	80.99 ± 0.16 **1.54↑**	86.19 ± 0.23 **13.96↑**

Empirical Findings: Large Graphs

Table 4: Node classification results on large-scale graphs (%). * indicates our implementation, while other results are taken from [12, 71]. The top 1st, 2nd and 3rd results are highlighted. OOM means out of memory.

	ogbn-proteins	ogbn-arxiv	ogbn-products	pokec
	2,534	169,343	2,449,029	1,632,803
	561,252	1,166,243	61,859,140	30,622,564
	C-AUC↑	Accuracy↑	Accuracy↑	Accuracy↑
	83 ± 0.26	70.97 ± 0.41	OOM	OOM
	15 ± 0.64	71.23 ± 0.59	OOM	OOM
	61 ± 0.33	70.13 ± 0.55	73.55 ± 0.21	76.59 ± 0.25
	17 ± 0.45	70.88 ± 0.24	74.63 ± 0.29	75.92 ± 0.68
	58 ± 0.26	72.44 ± 0.28	OOM	OOM
	62 ± 0.33	72.32 ± 0.36	OOM	OOM
	18 ± 0.37	72.41 ± 0.40	82.00 ± 0.43	66.37 ± 0.94
	81 ± 0.42	72.76 ± 0.29	82.27 ± 0.56	72.64 ± 0.67
	45 ± 1.15	59.90 ± 0.42	72.93 ± 0.13	71.00 ± 1.30
	86 ± 0.84	67.78 ± 0.28	73.96 ± 0.30	71.00 ± 1.30
	53 ± 0.38	72.63 ± 0.13	74.16 ± 0.31	73.76 ± 0.24
	92 ± 0.48	72.76 ± 0.33	81.54 ± 0.43	82.44 ± 0.76
	97 ± 0.47	73.46 ± 0.16	83.82 ± 0.11	86.10 ± 0.05
	53 ± 0.67	73.40 ± 0.22	83.82 ± 0.11	86.06 ± 0.25
	51 ± 0.35	71.74 ± 0.29	75.64 ± 0.21	75.45 ± 0.17
	29 ± 0.46 **4.78↑**	73.53 ± 0.12 **1.79↑**	82.33 ± 0.19 **6.69↑**	86.33 ± 0.17 **10.88↑**
	.68 ± 0.20	71.49 ± 0.27	78.29 ± 0.16	75.63 ± 0.38
	82.21 ± 0.32 **4.53↑**	73.00 ± 0.28 **1.51↑**	83.89 ± 0.36 **5.60↑**	85.97 ± 0.21 **10.34↑**
GAT	72.02 ± 0.44	71.95 ± 0.36	79.45 ± 0.59	72.23 ± 0.18
GAT*	85.01 ± 0.46 **12.99↑**	73.30 ± 0.18 **1.35↑**	80.99 ± 0.16 **1.54↑**	86.19 ± 0.23 **13.96↑**

Observations on Large Graphs.

Our implementation has significantly enhanced the previously reported results of classic GNNs, with some cases showing double-digit increases in accuracy. It has achieved the best results across these large graph datasets, either homophilous or heterophilous, and has outperformed state-of-the-art graph transformers.

Ablation Studies - Normalization

- **Normalization (either BN or LN) is important for node classification on large-scale graphs but less significant on smaller-scale graphs.**
  - The ablation of normalization **does not lead to substantial deviations on small graphs** (typically smaller than 1% difference).
  - The ablation results in significant accuracy reductions on large graphs (e.g., 4.69% and 4.79% for GAT* and GraphSAGE* respectively on ogbn-proteins).

Table 7: Ablation study on large-scale graphs (%).

	ogbn-proteins	ogbn-arxiv	ogbn-products	pokec
Metric	ROC-AUC↑	Accuracy↑	Accuracy↑	Accuracy↑
GCN*	**77.29** ± 0.46	**73.53** ± 0.12	**82.33** ± 0.19	**86.33** ± 0.17
(-) Normalization	74.48 ± 1.13	71.53 ± 0.14	80.01 ± 0.48	85.21 ± 0.23
(-) Dropout	74.85 ± 0.87	72.06 ± 0.13	79.30 ± 0.37	84.47 ± 0.38
(-) Residual Connections	73.19 ± 1.46	72.91 ± 0.17	-	79.59 ± 0.97
GraphSAGE*	**82.21** ± 0.32	**73.00** ± 0.28	**83.89** ± 0.36	**85.97** ± 0.21
(-) Normalization	77.42 ± 0.98	71.13 ± 0.27	82.12 ± 0.31	84.95 ± 0.33
(-) Dropout	80.52 ± 0.49	71.30 ± 0.21	80.36 ± 0.43	83.06 ± 0.28
(-) Residual Connections	81.75 ± 0.53	72.22 ± 0.49	-	85.81 ± 0.45
GAT*	**85.01** ± 0.46	**73.30** ± 0.18	**80.99** ± 0.16	**86.19** ± 0.23
(-) Normalization	80.32 ± 0.83	71.33 ± 0.29	78.62 ± 0.33	84.63 ± 0.64
(-) Dropout	82.48 ± 0.34	71.68 ± 0.32	77.68 ± 0.21	85.12 ± 0.49
(-) Residual Connections	82.43 ± 0.75	72.47 ± 0.34	-	81.37 ± 0.87

Ablation Studies - Dropout

- **Dropout is consistently found to be essential for node classification.**
  - The ablation of dropout leads to **substantial performance decreases in both homophilous and heterophilous graphs** (e.g., 4.28% decrease for GraphSAGE* on Amazon-Ratings and a 6.57% decrease on Roman-Empire).
  - This trend persists in **large-scale datasets**, where the ablation of dropout leads to **1-3% performance difference** for the three classic GNNs.

Table 7: Ablation study on large-scale graphs (%).

	ogbn-proteins	ogbn-arxiv	ogbn-products	pokec
Metric	ROC-AUC↑	Accuracy↑	Accuracy↑	Accuracy↑
GCN*	**77.29** ± 0.46	**73.53** ± 0.12	**82.33** ± 0.19	**86.33** ± 0.17
(-) Normalization	74.48 ± 1.13	71.53 ± 0.14	80.01 ± 0.48	85.21 ± 0.23
(-) Dropout	74.85 ± 0.87	72.06 ± 0.13	79.30 ± 0.37	84.47 ± 0.38
(-) Residual Connections	73.19 ± 1.46	72.91 ± 0.17	-	79.59 ± 0.97
GraphSAGE*	**82.21** ± 0.32	**73.00** ± 0.28	**83.89** ± 0.36	**85.97** ± 0.21
(-) Normalization	77.42 ± 0.98	71.13 ± 0.27	82.12 ± 0.31	84.95 ± 0.33
(-) Dropout	80.52 ± 0.49	71.30 ± 0.21	80.36 ± 0.43	83.06 ± 0.28
(-) Residual Connections	81.75 ± 0.53	72.22 ± 0.49	-	85.81 ± 0.45
GAT*	**85.01** ± 0.46	**73.30** ± 0.18	**80.99** ± 0.16	**86.19** ± 0.23
(-) Normalization	80.32 ± 0.83	71.33 ± 0.29	78.62 ± 0.33	84.63 ± 0.64
(-) Dropout	82.48 ± 0.34	71.68 ± 0.32	77.68 ± 0.21	85.12 ± 0.49
(-) Residual Connections	82.43 ± 0.75	72.47 ± 0.34	-	81.37 ± 0.87

Table 6: Ablation study on heterophilous graphs (%).

	Squirrel	Chameleon	Amazon-Ratings	Roman-Empire	Minesweeper	Questions
Metric	Accuracy↑	Accuracy↑	Accuracy↑	Accuracy↑	ROC-AUC↑	ROC-AUC↑
GCN*	**45.01** ± 1.63	**46.29** ± 3.40	**53.80** ± 0.60	**91.27** ± 0.20	**97.86** ± 0.24	**79.02** ± 0.60
(-) Normalization	44.13 ± 2.03	-	53.68 ± 0.82	90.53 ± 0.33	96.94 ± 1.96	-
(-) Dropout	42.89 ± 1.28	45.28 ± 4.78	51.37 ± 0.34	85.10 ± 0.61	94.28 ± 2.29	76.58 ± 0.40
(-) Residual Connections	43.14 ± 1.82	-	51.14 ± 0.34	74.84 ± 0.62	86.45 ± 0.89	75.87 ± 4.47
GraphSAGE*	**40.78** ± 1.47	**44.81** ± 4.74	**55.40** ± 0.21	**91.06** ± 0.27	**97.77** ± 0.62	**77.21** ± 1.28
(-) Normalization	40.27 ± 2.27	44.02 ± 3.53	54.41 ± 0.30	90.58 ± 0.24	97.64 ± 0.41	76.17 ± 0.41
(-) Dropout	38.83 ± 1.94	43.11 ± 3.36	51.12 ± 0.66	84.49 ± 0.35	93.83 ± 0.38	76.36 ± 1.50
(-) Residual Connections	40.06 ± 2.31	41.85 ± 3.86	53.52 ± 0.19	-	96.64 ± 0.85	-
GAT*	**41.73** ± 2.07	**44.13** ± 4.17	**55.54** ± 0.51	**90.63** ± 0.14	**97.73** ± 0.73	**77.95** ± 0.51
(-) Normalization	41.08 ± 1.63	43.25 ± 3.84	54.85 ± 0.39	89.69 ± 0.39	97.42 ± 0.85	76.32 ± 0.24
(-) Dropout	39.81 ± 3.15	41.19 ± 2.36	51.48 ± 0.28	82.47 ± 0.70	92.26 ± 4.63	76.19 ± 0.88
(-) Residual Connections	38.46 ± 1.96	42.57 ± 3.66	51.08 ± 0.49	85.15 ± 0.82	92.83 ± 1.61	75.17 ± 0.71

Ablation Studies – Residual connections

- **Residual connections exhibit more pronounced effect on large-scale graphs and heterophilous graphs than on homophilous graphs.**
  - The ablation of residual connections leads to **substantial performance decreases for GCN* and GAT* on large-scale graphs** such as ogbn-proteins and pokec.
  - The effect is even **more dramatic on heterophilous graphs**, with the classic GNNs exhibiting the most significant accuracy reduction on Roman-Empire, for instance, a **24.43% for GCN* and 25.48% for GAT***.

Table 7: Ablation study on large-scale graphs (%).

	ogbn-proteins	ogbn-arxiv	ogbn-products	pokec
Metric	ROC-AUC↑	Accuracy↑	Accuracy↑	Accuracy↑
GCN*	**77.29** ± 0.46	**73.53** ± 0.12	**82.33** ± 0.19	**86.33** ± 0.17
(-) Normalization	74.48 ± 1.13	71.53 ± 0.14	80.01 ± 0.48	85.21 ± 0.23
(-) Dropout	74.85 ± 0.87	72.06 ± 0.13	79.30 ± 0.37	84.47 ± 0.38
(-) Residual Connections	73.19 ± 1.46	72.91 ± 0.17	-	79.59 ± 0.97
GraphSAGE*	**82.21** ± 0.32	**73.00** ± 0.28	**83.89** ± 0.36	**85.97** ± 0.21
(-) Normalization	77.42 ± 0.98	71.13 ± 0.27	82.12 ± 0.31	84.95 ± 0.33
(-) Dropout	80.52 ± 0.49	71.30 ± 0.21	80.36 ± 0.43	83.06 ± 0.28
(-) Residual Connections	81.75 ± 0.53	72.22 ± 0.49	-	85.81 ± 0.45
GAT*	**85.01** ± 0.46	**73.30** ± 0.18	**80.99** ± 0.16	**86.19** ± 0.23
(-) Normalization	80.32 ± 0.83	71.33 ± 0.29	78.62 ± 0.33	84.63 ± 0.64
(-) Dropout	82.48 ± 0.34	71.68 ± 0.32	77.68 ± 0.21	85.12 ± 0.49
(-) Residual Connections	82.43 ± 0.75	72.47 ± 0.34	-	81.37 ± 0.87

Table 6: Ablation study on heterophilous graphs (%).

	Squirrel	Chameleon	Amazon-Ratings	Roman-Empire	Minesweeper	Questions
Metric	Accuracy↑	Accuracy↑	Accuracy↑	Accuracy↑	ROC-AUC↑	ROC-AUC↑
GCN*	**45.01** ± 1.63	**46.29** ± 3.40	**53.80** ± 0.60	**91.27** ± 0.20	**97.86** ± 0.24	**79.02** ± 0.60
(-) Normalization	44.13 ± 2.03	-	53.68 ± 0.82	90.53 ± 0.33	96.94 ± 1.96	-
(-) Dropout	42.89 ± 1.28	45.28 ± 4.78	51.37 ± 0.34	85.10 ± 0.61	94.28 ± 2.29	76.58 ± 0.40
(-) Residual Connections	43.14 ± 1.82	-	51.14 ± 0.34	74.84 ± 0.62	86.45 ± 0.89	75.87 ± 4.47
GraphSAGE*	**40.78** ± 1.47	**44.81** ± 4.74	**55.40** ± 0.21	**91.06** ± 0.27	**97.77** ± 0.62	**77.21** ± 1.28
(-) Normalization	40.27 ± 2.27	44.02 ± 3.53	54.41 ± 0.30	90.58 ± 0.24	97.64 ± 0.41	76.17 ± 0.41
(-) Dropout	38.83 ± 1.94	43.11 ± 3.36	51.12 ± 0.66	84.49 ± 0.35	93.83 ± 0.38	76.36 ± 1.50
(-) Residual Connections	40.06 ± 2.31	41.85 ± 3.86	53.52 ± 0.19	-	96.64 ± 0.85	-
GAT*	**41.73** ± 2.07	**44.13** ± 4.17	**55.54** ± 0.51	**90.63** ± 0.14	**97.73** ± 0.73	**77.95** ± 0.51
(-) Normalization	41.08 ± 1.63	43.25 ± 3.84	54.85 ± 0.39	89.69 ± 0.39	97.42 ± 0.85	76.32 ± 0.24
(-) Dropout	39.81 ± 3.15	41.19 ± 2.36	51.48 ± 0.28	82.47 ± 0.70	92.26 ± 4.63	76.19 ± 0.88
(-) Residual Connections	38.46 ± 1.96	42.57 ± 3.66	51.08 ± 0.49	85.15 ± 0.82	92.83 ± 1.61	75.17 ± 0.71

Ablation Studies – Number of layers

- **Deeper networks generally lead to greater performance gains on heterophilous graphs compared to homophilous graphs.**
  - The **performance trends for GCN* and GraphSAGE* are consistent** across different graph types.
  - On homophilous graphs and ogbn-arxiv, both models achieve **optimal performance** with a range of **2 to 6 layers.**
  - In contrast, **on heterophilous graphs**, their performance improves with an increasing number of layers, indicating that **deeper networks are beneficial** for these graphs.

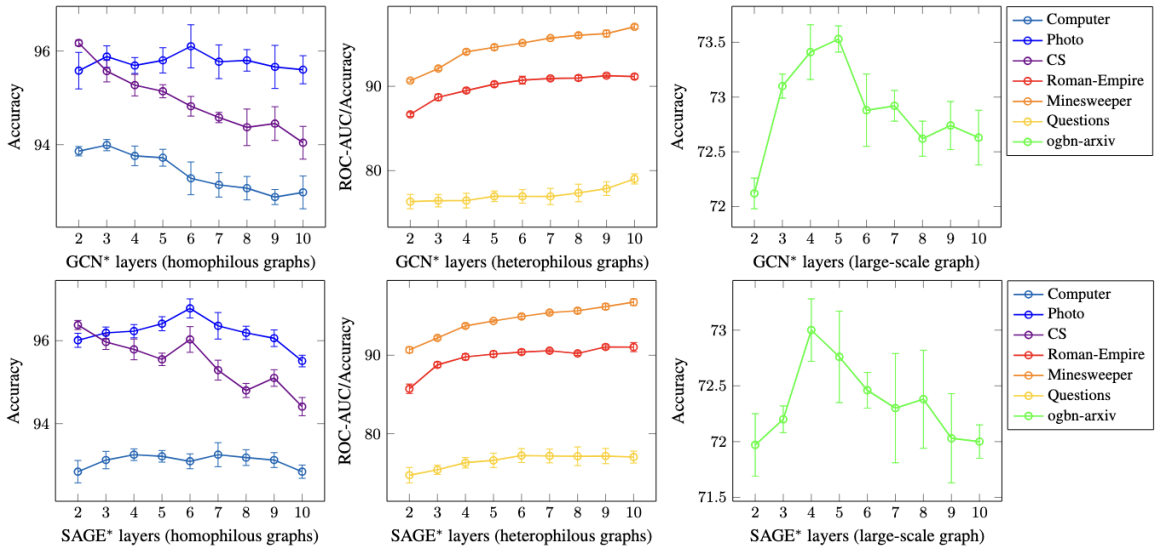


Figure 1: Ablation studies of the number of layers showing, from left to right, results for homophilous graphs, heterophilous graphs, and large-scale graphs, respectively.

Conclusions

- With proper hyperparameter tuning, classic GNNs can achieve highly competitive performance in node classification across homophilous and heterophilous graphs with up to millions of nodes.
- Notably, classic GNNs outperform state-of-the-art GTs, achieving the top rank on 17 out of 18 datasets.

https://github.com/LUOyk1999/tunedGNN

Conclusions

- With proper hyperparameter tuning, classic GNNs can achieve highly competitive performance in node classification across homophilous and heterophilous graphs with up to millions of nodes.
- Notably, classic GNNs outperform state-of-the-art GTs, achieving the top rank on 17 out of 18 datasets.

Thanks for listening!

https://github.com/LUOyk1999/tunedGNN