SIGMA: An Efficient Heterophilous Graph Neural Network with Fast Global Aggregation

Haoyu Liu, Ningyi Liao and Siqiang Luo

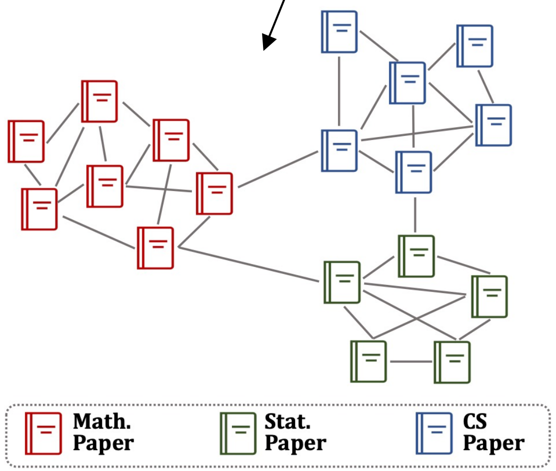
Nanyang Technological University

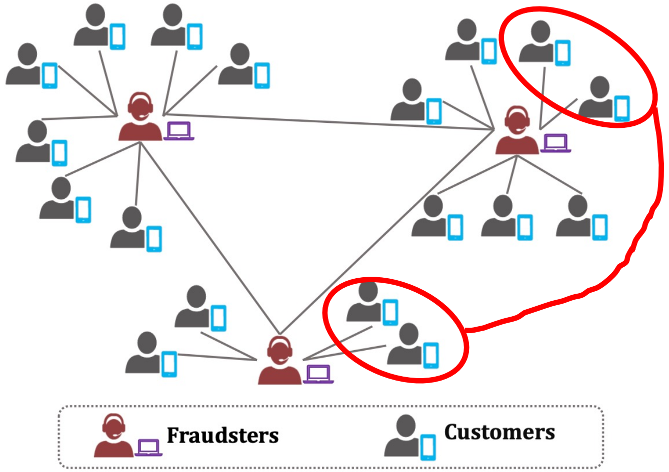
Presenter : Haoyu Liu

21 MAY 2025

Graph Learning from Homophily to Heterophily.

- Traditional GNNs often aggregate features from local neighbors hop-by-hop, relying on the *homophily* assumption that near neighbors are of intra-class.





- Graphs often meet *heterophily*, i.e., connected nodes are of inter-class, where traditional GNNs suffer due to mixture of inter-class nodes.

- One effective solution tends to leverage global homophily – nodes with distance in the whole graph range within intra-class.

Leverage Global Information into heterophily GNNs.

- *Feature Similarity*:
  calculate cosine similarity for all node pairs' embedding and select the relevant ones.

- Useful while requires $O(n^2)$ calculation cost.

- *Learned Global Homophily*: iteratively calculating and learning homophily correlation based on local structure for all node pairs.

- Complicated learning process.
- Expensive aggregation complexity, linear to the number of edges $O(k_1 m)$.

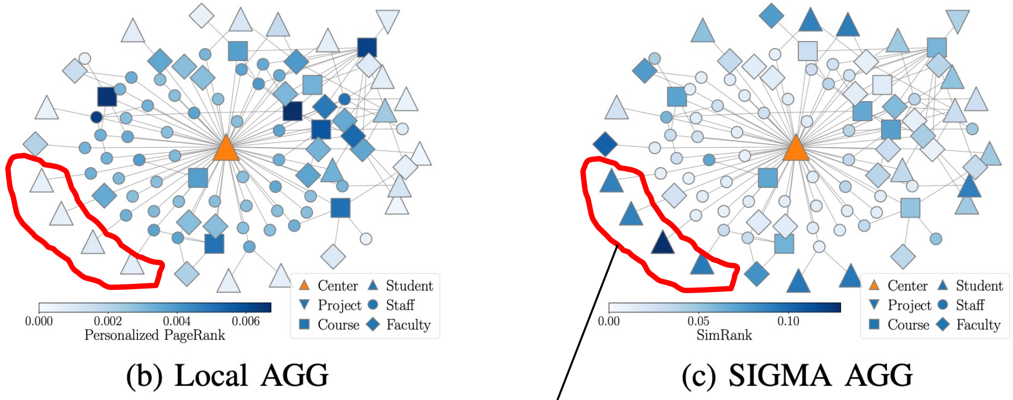
- Can we grasp *effective global homophily* with *superior efficiency?*

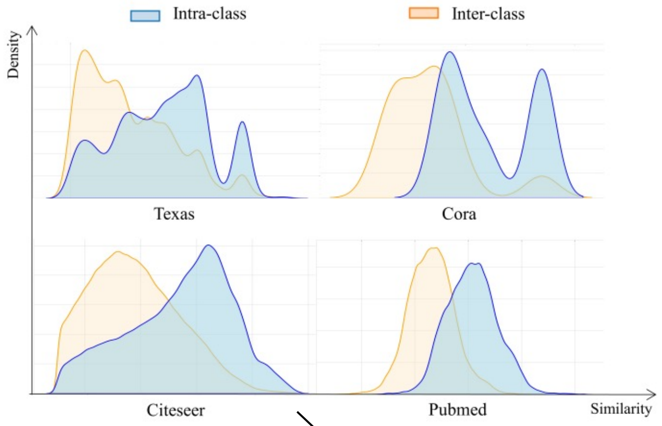
- The answer is SimRank!

SimRank – Global Similarity with Great Homophily Patterns.

- SimRank is a measure of node pair similarity based on graph topology, with the intuition that *two nodes are similar if they are connected by similar neighbors*.

- Empirically, it succeeds in assigning higher values for nodes of intra-class.





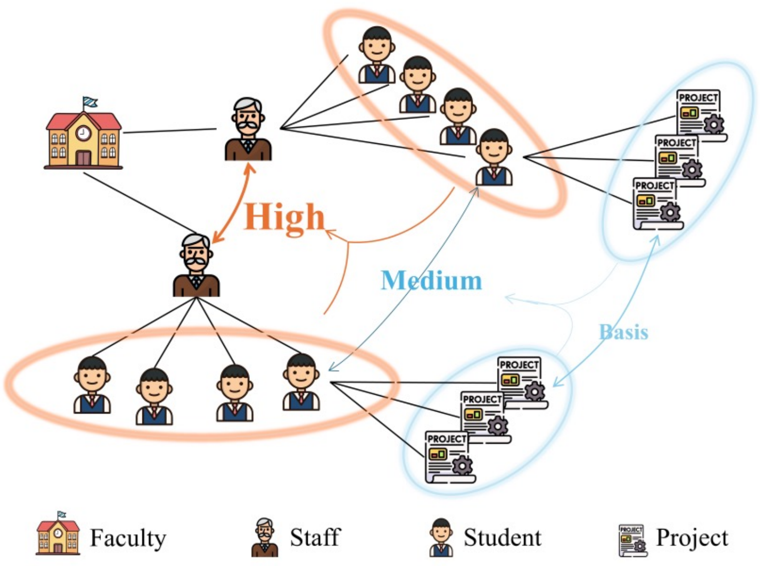
Detecting more homophily nodes

Intra-class node pairs achieve higher SimRank scores

- Why is SimRank effective?

Why SimRank Works – intuitive understanding.

- SimRank can be calculated as: $\mathbf{S}(u,v) = \frac{c}{|N_u||N_v|} \sum_{u',v' \in N_u, N_v} \mathbf{S}(u',v')$



(a) Intuitive Example of Global Similarity

The two nodes representing staffs' websites are more likely to be considered similar and assigned the same label because of their similar neighbors, i.e., respective students.

Likewise, student nodes are similar if connecting to similar projects. By this means, it establishes distinct relationships for similar nodes even though they are not directly connected.

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Why SimRank Works – perspective from *pairwise random walk*.

- *Pairwise random walk*, measures the probability that two random walks starting from node u and v simultaneously and meet at the same node w for the first time.

- The probability of such random walk pairs for all tours $t^{(2\ell)}$ with length 2ℓ can be formulated as:

$$\overleftrightarrow{P}(u, v|t^{(2\ell)}) = \sum_{t^{(2\ell)}} p(x|u, t^{(\ell)}_{u:x}) \cdot p(x|v, t^{(\ell)}_{v:x})$$
$$= \sum_{w\in V} p(w|u, t^{(\ell)}_{u:w}) \cdot p(w|v, t^{(\ell)}_{v:w})$$

Generally, a higher probability of such walks indicates strong connectivity between the source and end nodes.

- Re-formulate the SimRank-based aggregation as follow:

$$\widehat{\boldsymbol{Z}} = \boldsymbol{S}\boldsymbol{H} \longrightarrow \widehat{\boldsymbol{Z}}_u = \sum_{\ell=1}^{\infty} c^{\ell} \sum_{v\in V} \overleftrightarrow{P}(u, v|t^{(2\ell)}) \cdot \boldsymbol{H}_v.$$

SimRank-based aggregation is global, by counting on *all possible tours* connecting each node pair (u, v).

Why SimRank Works – perspective from *pairwise random walk*.

- The *probability* of nodes u and v *being homophily,* denoted as H_p^ℓ is calculated as:

$$\begin{aligned} H_p^\ell &= p^2 \cdot H_p^{\ell-1} + (1-p)^2 \cdot H_p^{\ell-1} \\ &= (2p^2 - 2p + 1) \cdot H_p^{\ell-1} = \ldots = \boxed{(2p^2 - 2p + 1)^\ell.} \end{aligned}$$

p denotes the heterophily extent.

- Derivative it over p, we get:

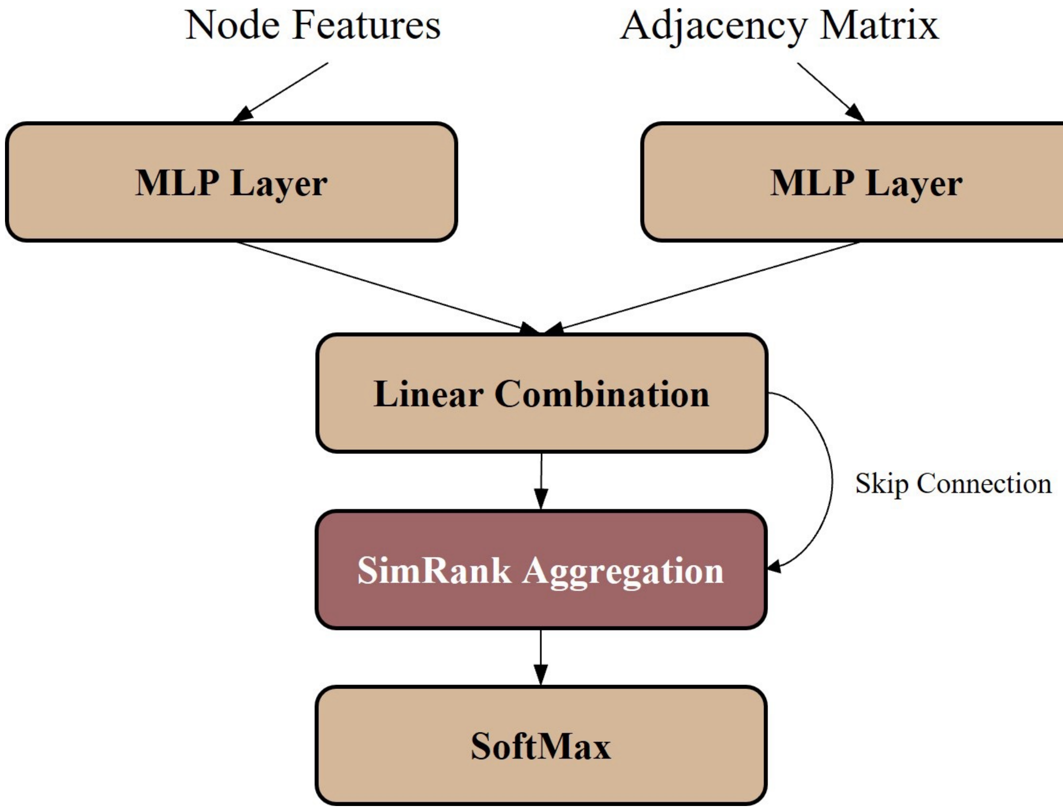
$$\frac{d}{dp} H_p^\ell = 2\ell \cdot (2p-1)\,(2p^2 - 2p + 1)^{\ell-1}\ .$$

- Under heterophily graph settings with $p > 0.5$ such the *derivative* $\Delta > 0$, we conclude:
  1) Larger ℓ results in a faster increase in the homophily probability.
  2) Δ increases rapidly as p grows, causing H_p^ℓ to approach one more quickly.

The probability of a central node being homophily rockets when graph becomes more heterophily.

SIGMA – an efficient heterophily GNN.

- SIGMA with simple and elegant design:



$$\mathbf{H}_{\mathbf{A}} = \texttt{MLP}_A(\mathbf{A}),\ \mathbf{H}_{\mathbf{X}} = \texttt{MLP}_X(\mathbf{X}),$$
$$\mathbf{H} = \texttt{MLP}_H\left(\delta \cdot \mathbf{H}_{\mathbf{X}} + (1-\delta) \cdot \mathbf{H}_{\mathbf{A}}\right).$$

$$\texttt{AGG}:\ \widehat{\mathbf{Z}}_u = \textstyle\sum_{v \in V} \mathbf{S}(u,v) \cdot \mathbf{H}_v,$$
$$\texttt{UPD}:\ \mathbf{Z}_{\mathbf{u}} = (1-\alpha) \cdot \widehat{\mathbf{Z}}_u + \alpha \cdot \mathbf{H}_u,$$

SIGMA – an efficient heterophily GNN.

- Superior calculation complexity:

Model	**Aggregation**	**Inference**
Geom-GCN [12]	$\mathcal{O}(n^2 f + mf)$	$\mathcal{O}(Ln^2 f + Lmf + nf^2)$
GPNN [13]	$\mathcal{O}(n^2 f^2 + nf)$	$\mathcal{O}(n^2 f^2 + Lmf + nf^2)$
U-GCN [15]	$\mathcal{O}(dmf + n^2 f + k_1 nf)$	$\mathcal{O}(dmf + n^2 f + k_1 nf + nf^2)$
WR-GAT [18]	$\mathcal{O}(Lmf + L\|R\|n^2 f + nf^2)$	$\mathcal{O}(L\|R\|n^2 f + mf + Lnf^2)$
GloGNN [16]	$\mathcal{O}(k_2 mfl_{norm})$	$\mathcal{O}(Lk_2 mfl_{norm} + mf + Lnf^2)$
SIGMA(ours)	$\boxed{\mathcal{O}(knf)}$	$\mathcal{O}(knf + mf + nf^2)$

- We improve previous aggregation time cost to linear of node size, i.e. $\underline{O(n)}$.

Experiments – classification accuracy

- Top ranking at **1.2** among 13 baselines, achieving the highest average accuracy on 9 out of 12 commonly used datasets.

Dataset	TEXAS	CITESEER	CORA	CHAMELEON	PUBMED	SQUIRREL	GENIUS	ARXIV	PENN94	TWITCH	SNAP	POKEC	
MLP	80.81±4.7	74.02±1.9	75.69±2.0	46.21±2.9	87.16±0.3	28.77±1.5	86.68±0.1	36.70±0.2	73.61±0.4	60.92±0.1	31.34±0.1	62.37±0.1	10.3
GAT	52.16±6.6	76.55±1.2	87.30±1.1	60.26±2.5	86.33±0.5	40.72±1.5	55.80±0.8	46.05±0.5	81.53±0.5	59.89±4.1	45.37±0.4	71.77±6.1	8.9
GBKGNN	81.08±4.8	79.18±0.9	87.29±0.4	61.59±2.3	89.11±0.2	55.90±1.1	OOM	OOM	OOM	OOM	OOM	OOM	8.8
HogGCN	85.17±4.4	76.15±1.7	87.04±1.1	67.27±1.6	88.79±0.4	58.26±1.5	OOM	OOM	OOM	OOM	OOM	OOM	8.7
WRGAT	83.62±5.5	76.81±1.8	88.20±2.2	65.24±0.8	88.52±0.9	48.85±0.7	OOM	OOM	74.32±0.5	OOM	OOM	OOM	8.5
H_2GCN	84.16±7.0	77.11±1.5	87.87±1.2	60.11±2.1	89.49±0.4	36.48±1.8	OOM	49.09±0.1	81.31±0.6	OOM	OOM	OOM	8.2
GPRGNN	78.38±4.3	77.13±1.6	87.95±1.2	46.58±1.7	87.54±0.4	31.61±1.2	90.05±0.3	45.07±0.2	81.38±0.2	61.89±0.3	40.19±0.1	78.83±0.1	8.0
GCN	55.14±5.1	76.50±1.3	86.98±1.2	64.82±2.2	88.42±0.5	53.43±2.0	87.42±0.3	46.02±0.2	82.47±0.2	62.18±0.2	45.65±0.0	75.45±0.1	7.6
ACMGCN	84.67±4.3	77.13±1.7	87.91±0.9	66.93±1.8	89.17±0.52	54.40±1.8	80.33±3.9	47.16±0.6	82.52±0.9	62.01±0.7	55.14±0.1	63.81±5.2	6.9
MixHop	77.84±7.7	76.26±1.3	87.61±0.8	60.50±2.5	85.31±0.6	43.80±1.4	90.58±0.1	51.81±0.1	83.47±0.7	65.64±0.2	52.16±0.1	81.07±0.1	6.6
GCNII	77.57±3.8	77.33±1.4	88.37±1.2	63.86±3.0	89.36±0.3	38.47±1.5	90.24±0.1	47.21±0.2	82.92±0.5	63.39±0.6	47.59±0.6	78.94±0.1	5.7
LINKX	74.60±8.3	73.19±0.9	84.64±1.1	68.42±1.3	87.86± 0.7	61.81±1.8	90.77±0.2	56.00±1.3	84.71±0.5	66.06±0.2	61.95±0.1	82.04±0.1	5.5
GloGNN	84.05±4.9	77.22±1.7	88.33±1.0	71.21±1.8	89.24±0.4	57.88±1.7	90.91±0.1	54.79±0.2	85.74±0.4	66.34±0.3	62.03±0.2	83.05±0.1	2.9
SIGMA	85.32±4.7	77.52±1.5	88.96±1.2	72.13±1.7	89.76±0.3	62.04±1.6	91.68±0.6	55.16±0.3	86.31±0.3	67.21±0.3	64.63±0.2	82.33±0.1	1.2

Experiments – efficiency

- SIGMA costs the least learning time over the six large-scale dataset, outperforms GloGNN by around **10**× and **5**× faster, on Penn94 and pokec correspondingly.

TABLE V: The average learning time (s) on large-scale datasets. We separately show the break down of precomputation (Pre.) and aggregation (AGG) when applicable. The overall learning efficiency are marked with first, second and third place.

Model	GENIUS			ARXIV			PENN94			TWITCH			SNAP			POKEC		
	Pre.	AGG	Learn	Pre.	AGG	Learn	Pre.	AGG	Learn	Pre.	AGG	Learn	Pre.	AGG	Learn	Pre.	AGG	Learn
LINKX	-	-	292.3	-	-	51.2	-	-	49.6	-	-	302.7	-	-	469.1	-	-	672.3
GloGNN	-	313.6	358.7	-	126.1	134.1	-	167.1	183.5	-	739.5	783.0	-	696.3	732.9	-	1417.8	1564.7
SIGMA	8.6	119.8	153.6	9.3	26.6	36.5	3.9	6.9	17.2	14.0	189.2	236.5	15.9	314.3	408.2	11.4	279.7	388.5

Conclusion

- We found that SimRank, with the re-formulation understanding from pairwise random walk, and empirical insights, is capable of grasping global graph homophily.

- Based on this, we design an efficient and effective heterophily graph neural network, named SIGMA, to address heterophily graph learning problems, with theoretical guarantee and superior computation complexity.

In future, we consider extend SIGMA to dynamic and attributed graphs.

THANKS!
Q&A