Distance Features, Labeling Tricks? Towards More Expressive GNNs

Pan Li
Assistant Professor
Department of Computer Science
Purdue University

Talk at LoGaG, 03/29/2022

Relevant Papers

1. Nested Graph Neural Networks, Zhang & Li, NeurIPS 2021

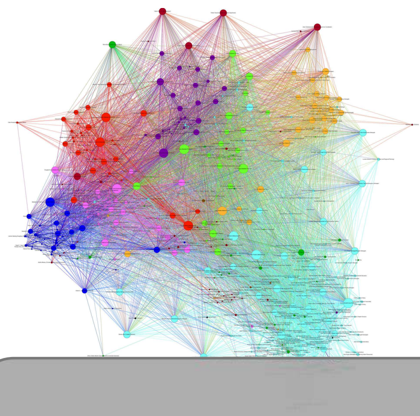
2. Labeling Trick: A Theory of Using Graph Neural Networks for Multi-Node Representation Learning, Zhang et al., NeurIPS 2021

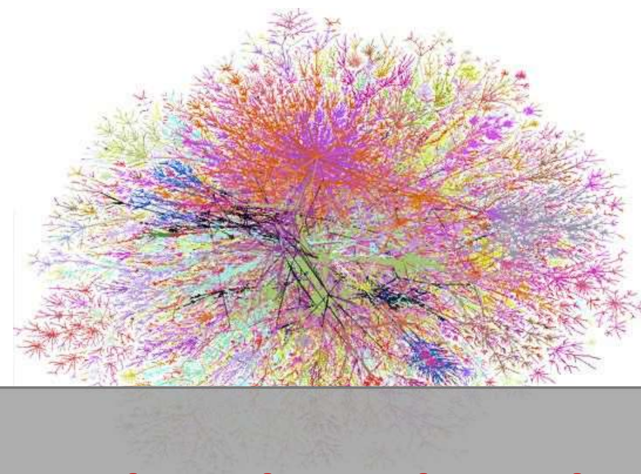
3. Distance Encoding: Design Provably more Powerful Neural Networks for Graph Representation Learning, Li et al., NeurIPS 2020

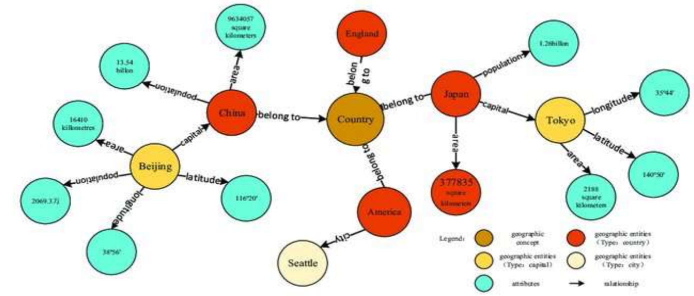
- From the perspective of the expressive power of GNNs: Transformers on graphs, Neural Bellman-Ford Networks, …

"Neural Bellman-Ford Networks: A General Graph Neural Network Framework for Link Prediction, " Zhu et al., NeurIPS 2021
"Do Transformers Really Perform Bad for Graph Representation?" Ying et al. NeurIPS 2021

Graph Structured Data is Everywhere…





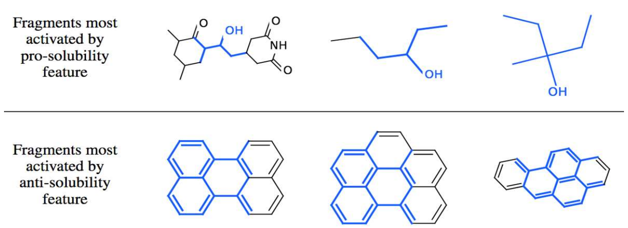


Social networks

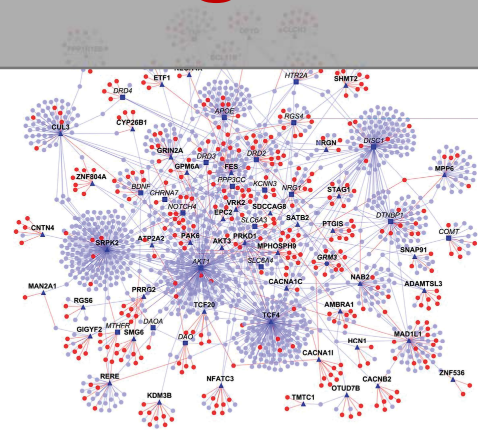
Internet networks

Knowledge graphs

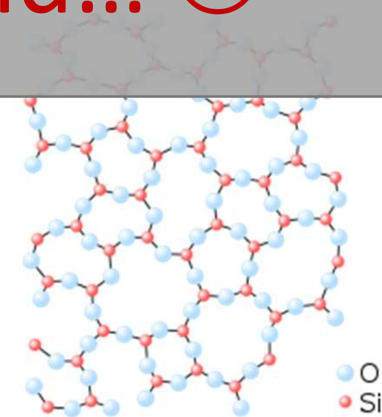
Let us skip the big background… ☺



Drug molecules



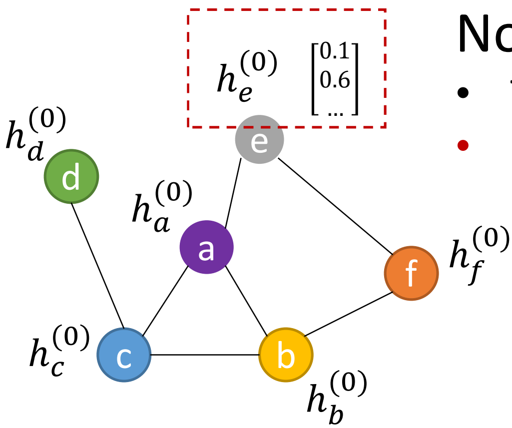
Protein-protein Interaction



Glass structure

Standard Graph Neural Networks

Graph Data (A, X): the adjacency matrix A, possibly with node attributes X.



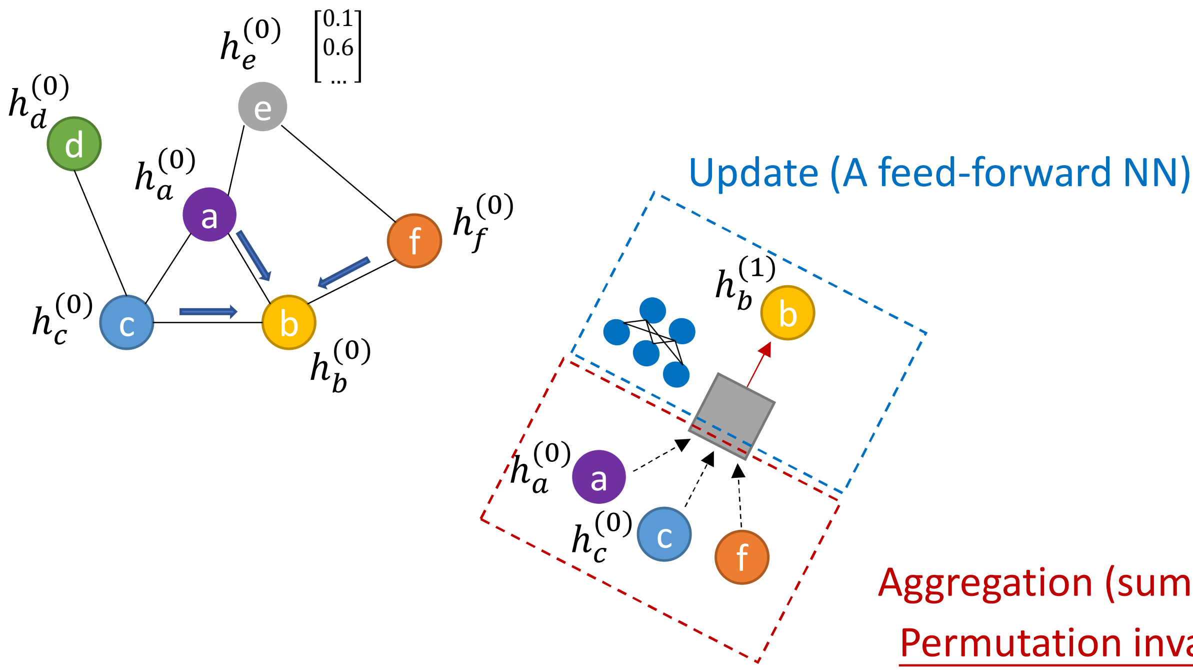
Node (feature) representation

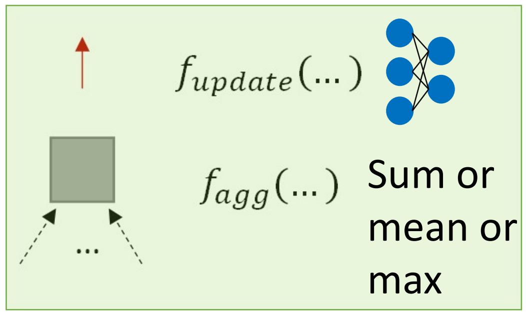
- Transformation of node attributes
- Constant if no node attributes are available

Do not consider using random features as for the slow convergence of their training procedure [Sato et al. SDM 2021][Abboud et al. IJCAI 2021]

Standard Graph Neural Networks

Graph Data (A, X): the adjacency matrix A, possibly with node attributes X.





Graph neural network: one layer

Aggregation (sum, mean, max pooling, attention, etc.)

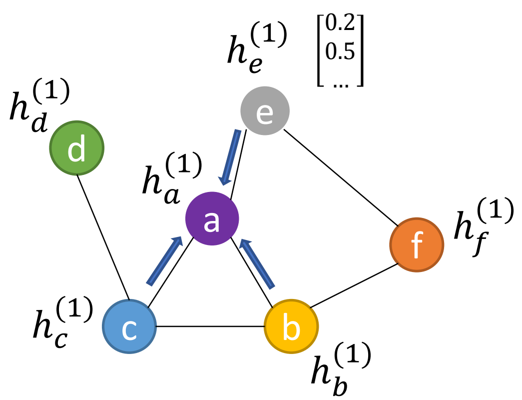
Permutation invariance w.r.t. the order of neighbors

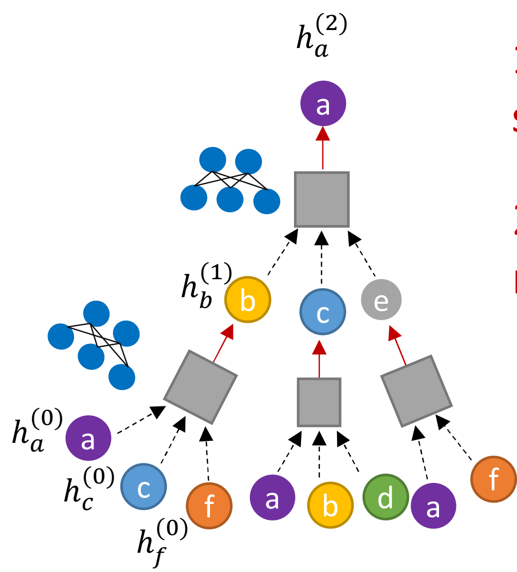
$$h_v^{(t+1)} = f_{update}\left(h_v^{(t)}, f_{agg}\left(\{h_u^{(t)} \middle| u \in N_v\}\right)\right),$$

where N_v denotes the set of the neighbors of node v.

Standard Graph Neural Networks

Graph Data (A, X): the adjacency matrix A, possibly with node attributes X.





Make prediction

1. [node level] Use node representations separately to predict node labels

2. [graph level] Aggregate all node representations to predict the graph label

$$h_v^{(t+1)} = f_{update}\left(h_v^{(t)}, f_{agg}\left(\{h_u^{(t)} \middle| u \in N_v\}\right)\right),$$

where N_v denotes the set of the neighbors of node v.

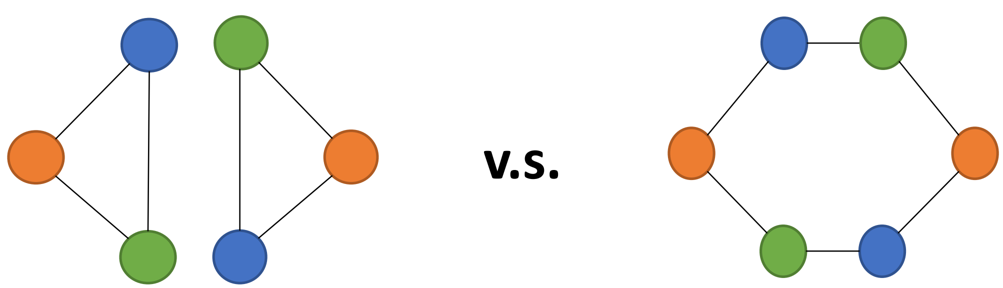
Topic for Today: The Expressive Power of GNNs

- **Function approximation**

  What class of functions can GNNs approximate?

- **Distinguishing graph structures**

  Whether can GNNs distinguish two different graph structures or not ?



Different node colors refer to different node attributes

Topic for Today: The Expressive Power of GNNs

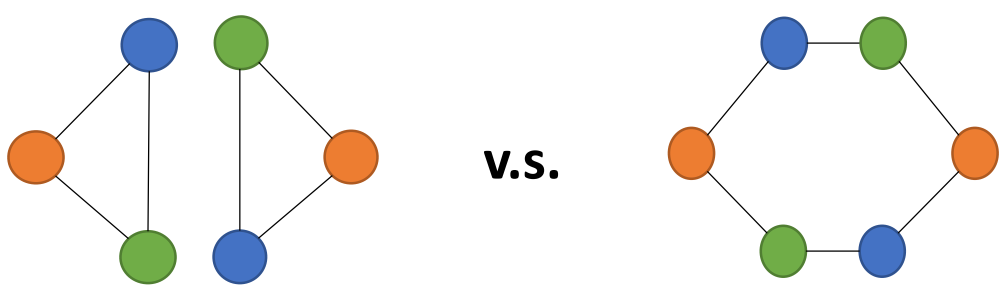
- Function approximation

  What class of functions can GNNs approximate?

More general but kind of equivalent under certain conditions [Chen et al., NeurIPS 2019][Azizian & Lelarge, ICLR 2021]

- Distinguishing graph structures [Our focus today]

  Whether can GNNs distinguish two different graph structures or not ?

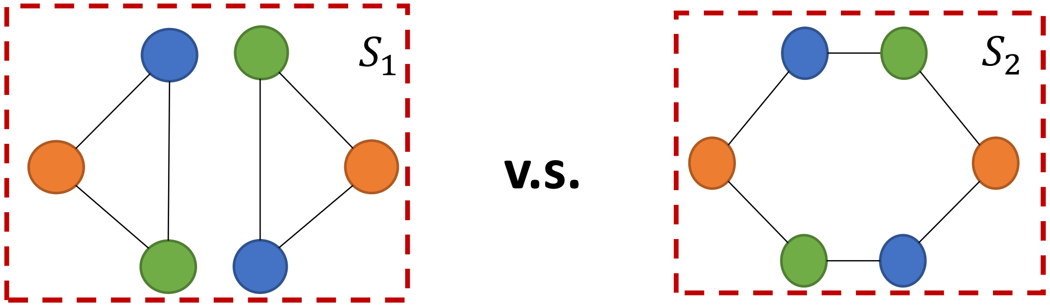


Different node colors refer to different node attributes

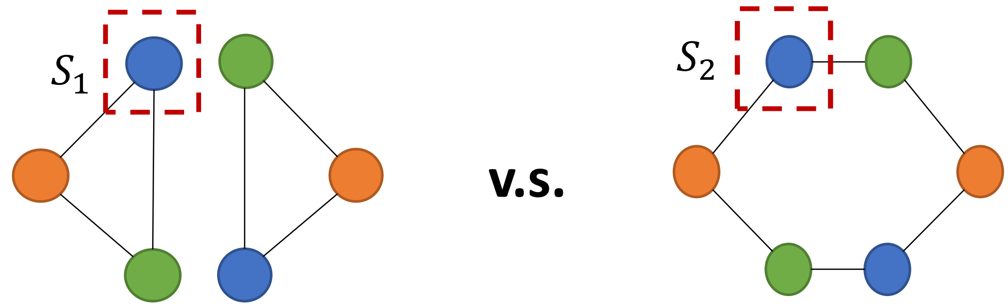
The Limited Expressive Power of Standard GNNs

The expressive power of standard GNNs is bounded by 1-Weisfeiler-Lehman test (1-WL test)

[graph level]



[node level]



If 1-WL test may not distinguish two graph-structured data, GNN cannot output representations that distinguish them.

--- [Xu et al., ICLR 2019] [Morris et al., AAAI 2019]

Clarification of Some Phrases

- Standard GNNs: Message passing, aggregate from directed neighbors of a node
  - Suffer from the 1-WL limitation

E.g., GCN, Graphsage, GIN, GAT, …

“Do Transformers Really Perform Bad for Graph Representation?” Ying et al. NeurIPS 2021

Clarification of Some Phrases

- Standard GNNs: Message passing, aggregate from directed neighbors of a node
  - Suffer from the 1-WL limitation

E.g., GCN, Graphsage, GIN, GAT, …

- Node representation refinement: $\{h_v^{(t)} | v \in V\} \rightarrow \{h_v^{(t+1)} | v \in V\}$
  - Standard GNNs adopt node rep. refinement
  - Some GNNs/procedures that adopt node rep. refinement are not standard GNNs, e.g., Graphomer (transformers on graphs)

[Ying et al. NeurIPS 2021]

  - Higher-order GNNs do not adopt node rep. refinement (not our focus)

[Maron et al. NeurIPS 2019]
[Morris et al. AAAI 2019]
[Chen et al. NeurIPS 2019]

"Do Transformers Really Perform Bad for Graph Representation?" Ying et al. NeurIPS 2021

Roadmap

- Distance features improve the expressive power of node representation refinement procedure [Zhang & Li, NeurIPS 2021]

- Labeling tricks enable (most expressive) node representation refinement for node-set representation learning [Zhang et al., NeurIPS 2021]

Nested Graph Neural Networks, Zhang & Li, NeurIPS 2021

Labeling Trick: A Theory of Using Graph Neural Networks for Multi-Node Representation Learning, Zhang et al., NeurIPS 2021

Roadmap

- **Distance features** improve the expressive power of node representation refinement procedure [Zhang & Li, NeurIPS 2021]

Distance encoding

[Li et al., NeurIPS 2020]

- **Labeling tricks** enable (most expressive) node representation refinement for node-set representation learning

[Zhang et al., NeurIPS 2021]

Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020

Roadmap

- Distance features improve the expressive power of node representation refinement procedure [Zhang & Li, NeurIPS 2021]

Distance encoding

[Li et al., NeurIPS 2020]

- Labeling tricks enable (most expressive) node representation refinement for node-set representation learning

[Zhang et al., NeurIPS 2021]

- Scalable Distance Encoding [Yin et al., submitted]

Roadmap

➢ Distance features improve the expressive power of node representation refinement procedure [Zhang & Li, NeurIPS 2021]

1. Definition for (general) distance features
2. Standard GNNs miss capturing distance features
3. How to use distance features in practice

   → node structural features, graph transformers

4. How powerful distance features could be in theory

Distance encoding

[Li et al., NeurIPS 2020]

Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020

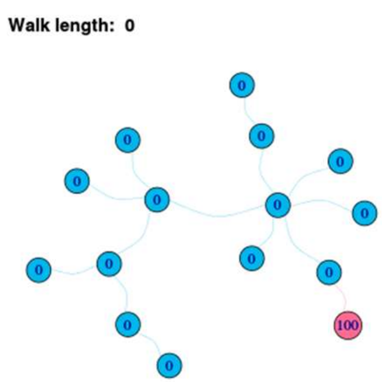
Nested Graph Neural Networks, Zhang & Li, NeurIPS 2021

Distance Features over Graphs

Distance features between two nodes

Given a graph A, the distance from node u to node v is $\zeta(v|A,u) = \left((W)_{uv}, (W^2)_{uv}, \dots, (W^k)_{uv}\right)$. W is the random walk matrix.

- Step k ~ O(the diameter of the graph) in theory; constant in practice



Landing times of random walks

Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020

Nested Graph Neural Networks, Zhang & Li, NeurIPS 2021

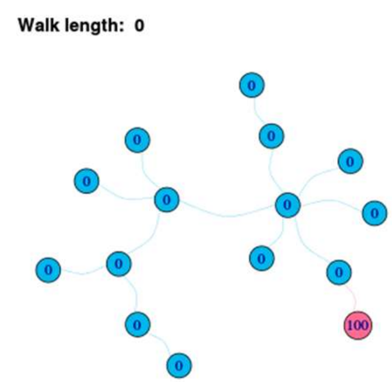
Distance Features over Graphs

Distance features between two nodes

Given a graph A, the distance from node u to node v is $\zeta(v|A,u) = \left((W)_{uv}, (W^2)_{uv}, \dots, (W^k)_{uv}\right)$. W is the random walk matrix.

- Step k ~ O(the diameter of the graph) in theory; constant in practice
- $\zeta(v|A,u)$ via a mapping g($\zeta(v|A,u)$) can represent many "distance measures" from u to v:

1. Shortest path distance between u and v,
2. Hitting time from u to v,
3. Personalized PageRank…
4. Generalized PageRank [Li et al., NeurIPS 2019]

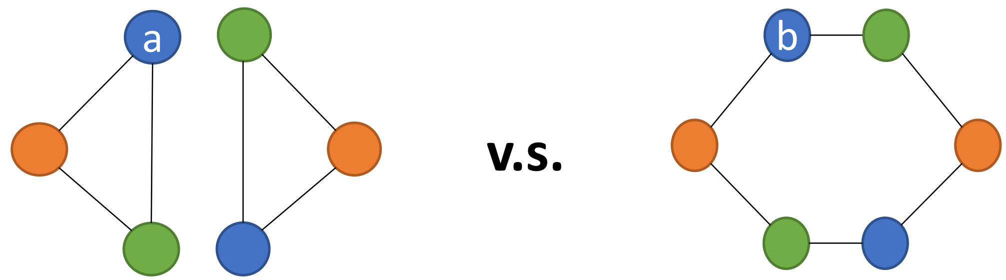


Landing times of Personalized Pagerank

Optimizing Generalized PageRank Methods for Seed-Expansion Community Detection, Li et al., NeurIPS 2019

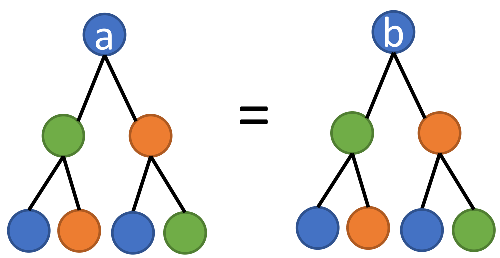
Standard GNNs Fail to Capture Distances

[node level]



Counting loops: Is node a (or node b) in a 3 loop?

---[Chen et al., NeurIPS 2020]

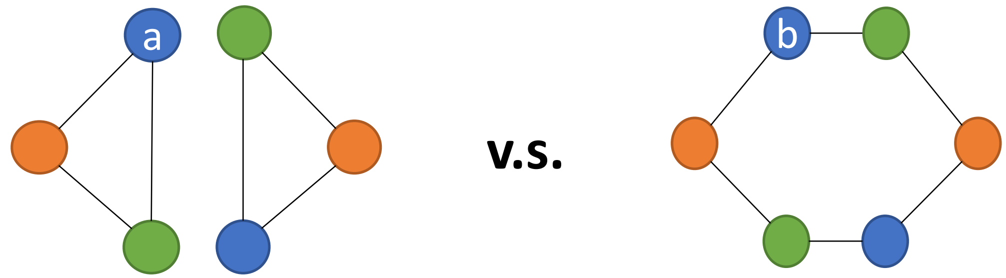


Standard GNNs cannot distinguish node a and node b.

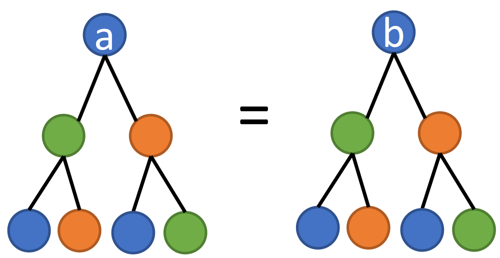
Can Graph Neural Networks Count Substructures? Chen et al., NeurIPS 2020

Standard GNNs Fail to Capture Distances

[node level]



Counting loops: Is node a (or node b) in a 3 loop?



Left: 3-step landing prob. $(W^3)_{aa} > 0$

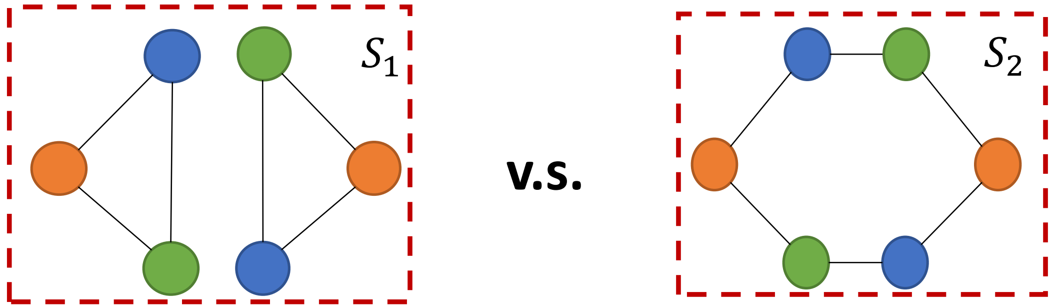
Right: 3-step landing prob. $(W^3)_{bb} = 0$

Standard GNNs cannot distinguish node a and node b.

Distance features address the question trivially.

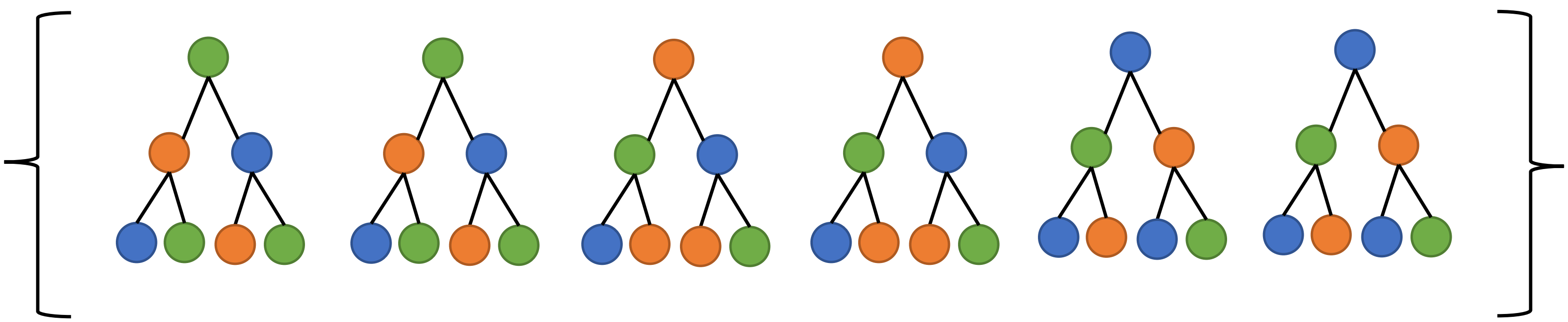
Standard GNNs Fail to Capture Distances

[graph level]



Distinguish two non-isomorphism graphs?

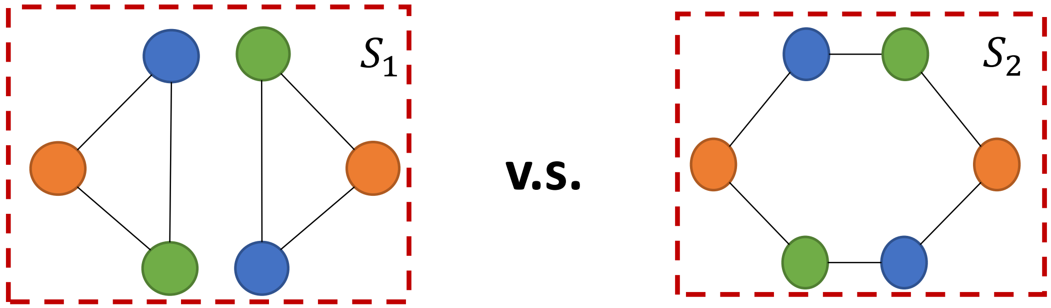
- Two graphs have the same the set of node representations.



Standard GNNs cannot distinguish the two graphs

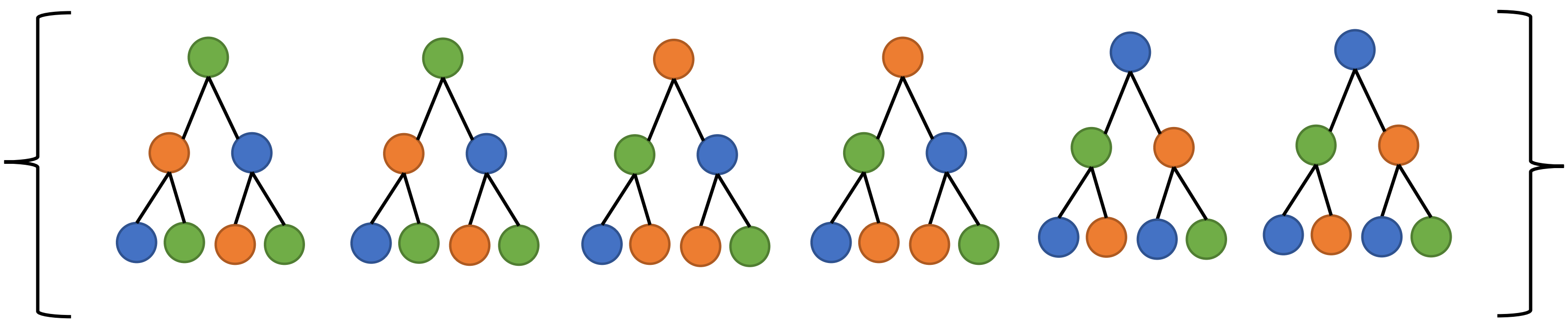
Standard GNNs Fail to Capture Distances

[graph level]



Distinguish two non-isomorphism graphs?

- Two graphs have the same the set of node representations.

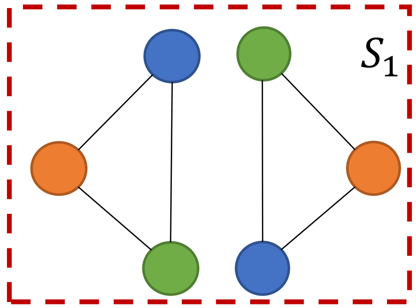


Standard GNNs cannot distinguish the two graphs

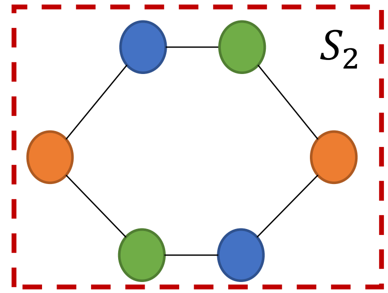
- Distance features can distinguish them trivially.
  - $(W^3)_{vv} > 0$ for every node v in the left graph
  - $(W^3)_{vv} = 0$ for every node v in the right graph

Standard GNNs Fail to Capture Distances

[graph level]



v.s.



Distinguish two non-isomorphism graphs?

- Standard GNNs/1-WL test cannot distinguish two regular graphs of the same size and node degree.

- Standard GNNs fail to distinguish two attributed-regular graphs.

---[Li & Leskovec, 2021]

Li & Leskovec, "The Expressive Power of Graph Neural Networks, " Chapter 5 in "Graph Neural Networks: Foundations, Frontiers, and Applications" edited by Wu et al., 2021

How to Use Distance Features?

- To get more expressive node representations
  - ➢ Use distance features as extra node attributes
  - ➢ Use distance features as extra edge attributes

How to Use Distance Features?

- To get more expressive node representations

  - Use distance features as extra node attributes

    1. "Identity-aware Graph Neural Networks, " You et al., AAAI 2021

    2. "Nested Graph Neural Networks, " Zhang & Li., NeurIPS 2021

    3. "From Stars to Subgraphs: Uplifting Any GNN with Local Structure Awareness, " Zhao et al., ICLR 2022

    ...

  - Use distance features as extra edge attributes

    1. "Do Transformers Really Perform Bad for Graph Representation?" Ying et al. NeurIPS 2021

    2. "MixHop: Higher-Order Graph Convolutional Architectures via Sparsified Neighborhood Mixing," Haija et al., ICML 2019

    ...

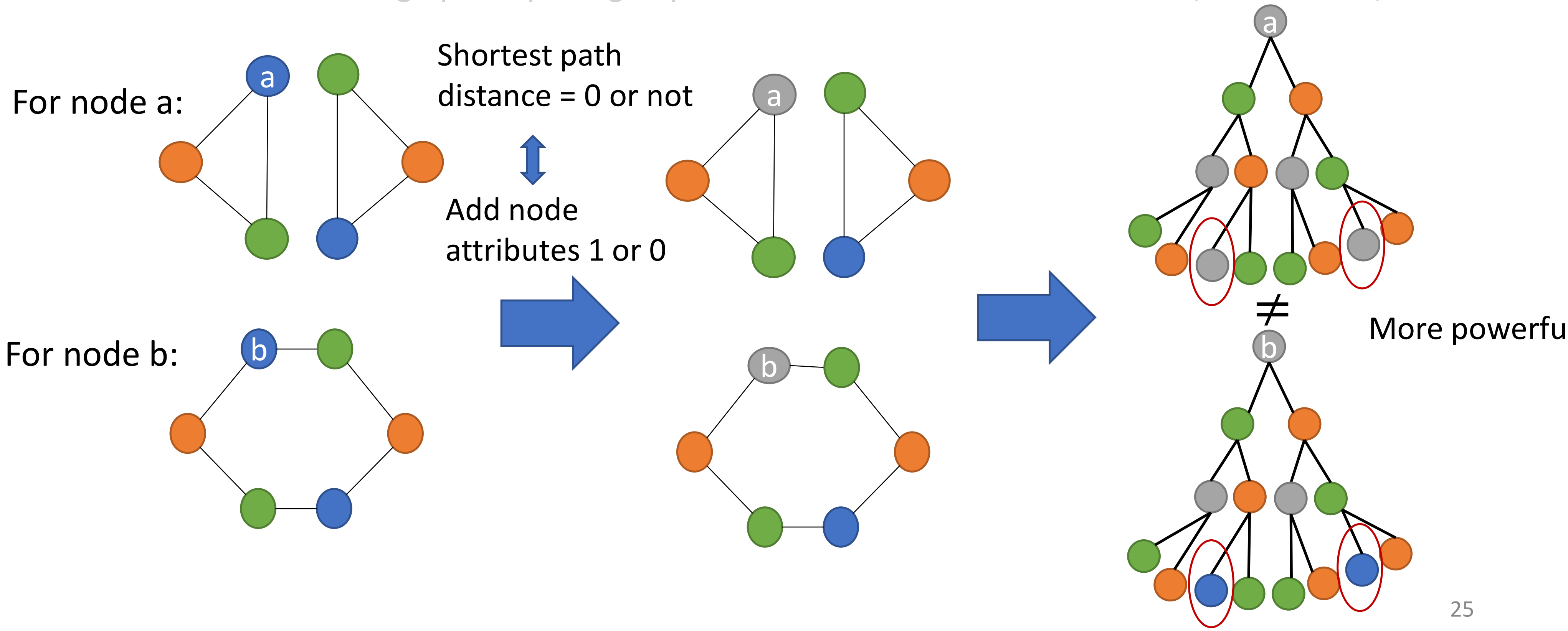
How to Use Distance Features?

- To get more expressive node representations
  - Use distance features as extra node attributes

“Identity-aware Graph Neural Networks, ” You et al., AAAI 2021

“Nested Graph Neural Networks, ” Zhang & Li., NeurIPS 2021

“From Stars to Subgraphs: Uplifting Any GNN with Local Structure Awareness, ” Zhao et al., ICLR 2022



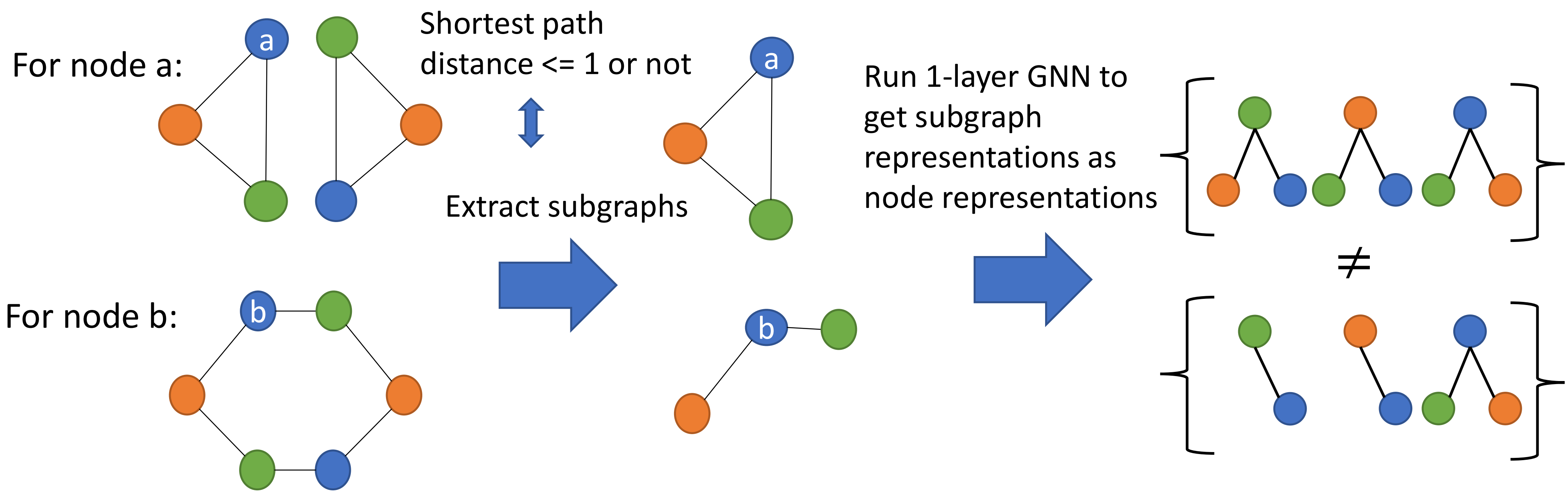
How to Use Distance Features?

- To get more expressive node representations
  - ➢ Use distance features as extra node attributes

“Identity-aware Graph Neural Networks,” You et al., AAAI 2021

“Nested Graph Neural Networks,” Zhang & Li., NeurIPS 2021

“From Stars to Subgraphs: Uplifting Any GNN with Local Structure Awareness,” Zhao et al., ICLR 2022



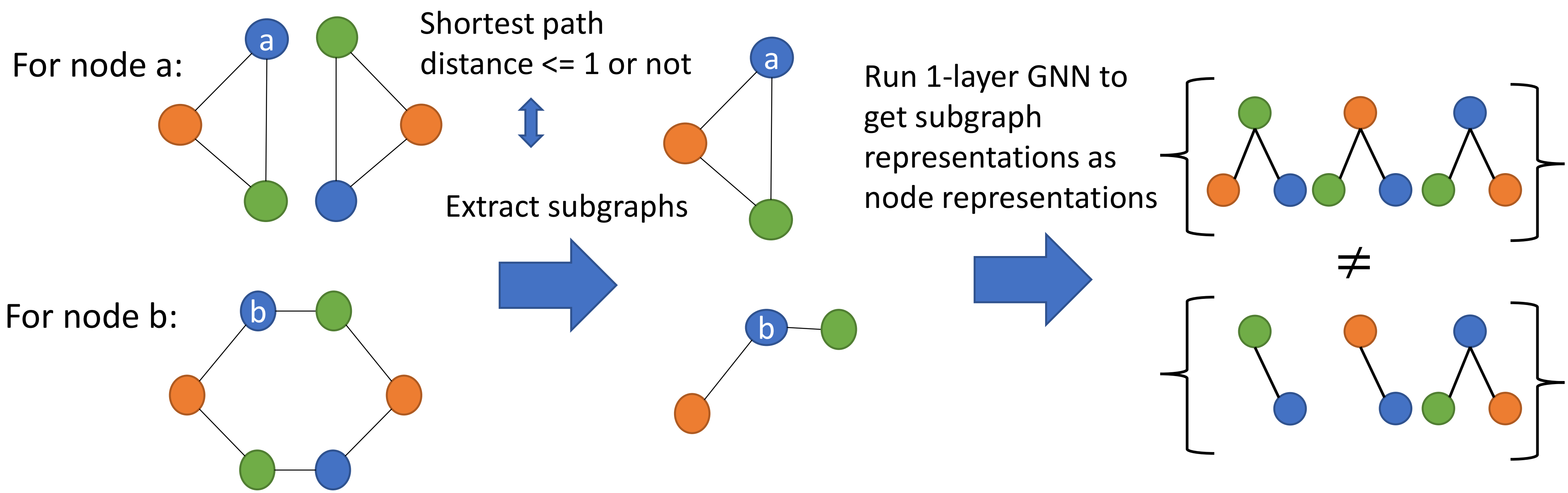
How to Use Distance Features?

- To get more expressive node representations
  - Use distance features as extra node attributes

“Identity-aware Graph Neural Networks, ” You et al., AAAI 2021

“Nested Graph Neural Networks, ” Zhang & Li., NeurIPS 2021

“From Stars to Subgraphs: Uplifting Any GNN with Local Structure Awareness, ” Zhao et al., ICLR 2022



How to Use Distance Features?

- To get more expressive node representations

  - Use distance features as extra node attributes

    1. “Identity-aware Graph Neural Networks, ” You et al., AAAI 2021

    2. “Nested Graph Neural Networks, ” Zhang & Li., NeurIPS 2021

    3. “From Stars to Subgraphs: Uplifting Any GNN with Local Structure Awareness, ” Zhao et al., ICLR 2022

  - Use distance features as extra edge attributes

    1. “Do Transformers Really Perform Bad for Graph Representation?” Ying et al. NeurIPS 2021

    2. “MixHop: Higher-Order Graph Convolutional Architectures via Sparsified Neighborhood Mixing,” Haija et al., ICML 2019

How to Use Distance Features?

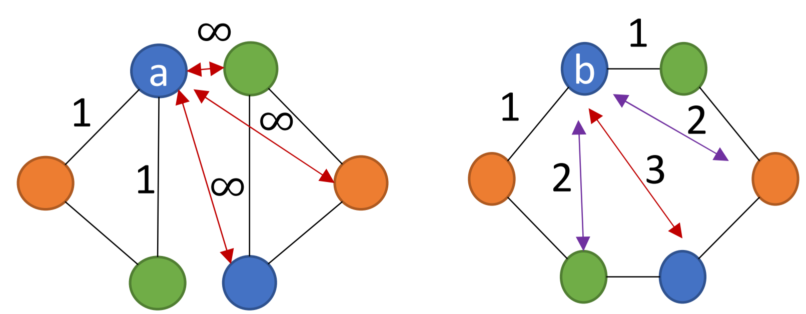
- To get more expressive node representations
  - ➢ Use distance features as extra edge attributes

“Do Transformers Really Perform Bad for Graph Representation?” Ying et al. NeurIPS 2021

“MixHop: Higher-Order Graph Convolutional Architectures via Sparsified Neighborhood Mixing,” Haija et al., ICML 2019

➢ Use distance features between two nodes to compute attention weight

Shortest path distance as edge features



Also check other transformer models for graphs [Dwivedi & Bresson, AAAI-DGL 2021]

How to Use Distance Features?

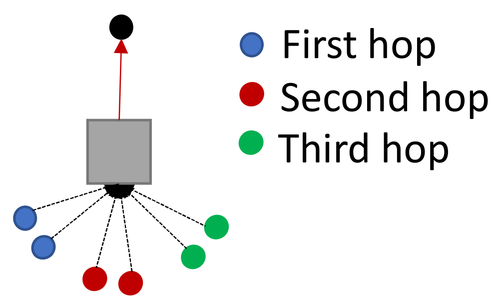
- To get more expressive node representations
  - Use distance features as extra edge attributes

“Do Transformers Really Perform Bad for Graph Representation?” Ying et al. NeurIPS 2021

“MixHop: Higher-Order Graph Convolutional Architectures via Sparsified Neighborhood Mixing,” Haija et al., ICML 2019

- Aggregate neighbors from multiple-hop neighborhood together within one GNN layer simultaneously
- Different hops use different parameters.

Shortest path distance (SPD) as edge features



$$h_{hop\ 0}^{(t+1)} \leftarrow \sigma(\underbrace{h_{hop\ 0}^{(t)} W_{hop\ 0}}_{\text{SPD = 0}} + \underbrace{h_{hop\ 1}^{(t)} W_{hop\ 1}}_{\text{SPD = 1}} + \underbrace{h_{hop\ 2}^{(t)} W_{hop\ 2}}_{\text{SPD = 2}} + \ \ldots)$$

How Powerful Distance Features are?

- Use distance features as extra node attributes

“Identity-aware Graph Neural Networks, ” You et al., AAAI 2021

“Nested Graph Neural Networks, ” Zhang & Li., NeurIPS 2021

“From Stars to Subgraphs: Uplifting Any GNN with Local Structure Awareness, ” Zhao et al., ICLR 2022

“Improving Graph Neural Network Expressivity via Subgraph Isomorphism Counting, ” Bouritsas et al. 2020

Provide the proof idea

Also generalizable to the case of using extra edge attributes

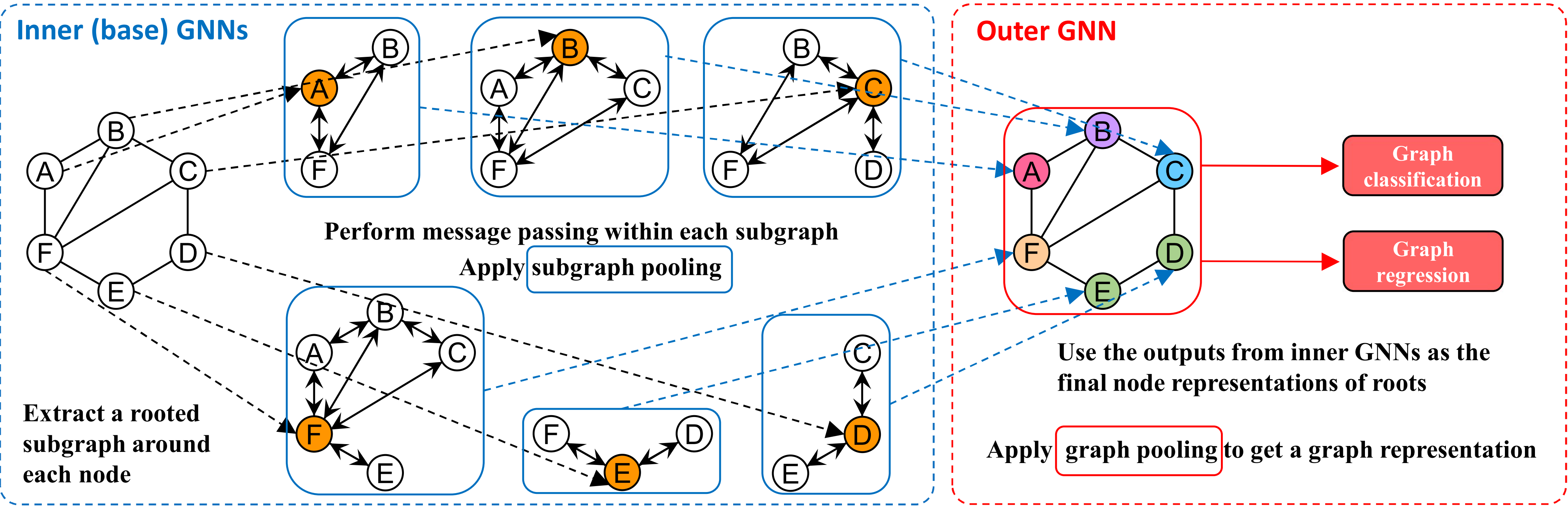
- Use distance features as extra edge attributes

“Do Transformers Really Perform Bad for Graph Representation?” Ying et al. NeurIPS 2021

“MixHop: Higher-Order Graph Convolutional Architectures via Sparsified Neighborhood Mixing,” Haija et al., ICML 2019

“Distance encoding: Design provably more powerful neural networks for graph representation learning, ” Li et al., NeurIPS 2020

Nested Graph Neural Networks



1. Extract the contextual subgraph (based on shortest path distance SPD h) around each node
2. Run a standard GNN on each subgraph and perform subgraph pooling to get the node representation
3. Pool node representations as the representation of the original graph

“Nested Graph Neural Networks,” Zhang & Li., NeurIPS 2021

How Powerful Distance Features are?

Theorem [Zhang, Li, NeurIPS'21]

(Condition)

Not crucial, can be generalized

Consider r-regular non-attributed n-sized graphs G, where $r \in [3, \sqrt{0.5 \log n}]$.

Distance info to be used

- Subgraph extraction based on SPD $h \in (0.5 \frac{\log n}{\log(r-1)}, 0.66 \frac{\log n}{\log(r-1)})$
- The standard GNN with at least $h - 0.5 \frac{\log n}{\log(r-1)}$ layers

$1 - o\left(\frac{1}{n^{0.5-\epsilon}}\right)$ based on a tighter analysis

(Conclusion)

- ➢ The nested GNN model can distinguish almost all $(1 - o(1))$ such graphs.
- ➢ However, standard GNNs cannot distinguish these graphs with even an infinite number of layers.

Empirical Results

Table 5: Results (%) on OGB datasets (* virtual node).

	ogbg-molhiv (AUC)		ogbg-molpcba (AP)	
Method	Validation	**Test**	Validation	**Test**
CCN*	83.84±0.91	75.99±1.19	24.95±0.42	24.24±0.34
GIN*	84.79±0.68	77.07±1.49	27.98±0.25	27.03±0.23
Deep LRP	82.09±1.16	77.19±1.40	–	–
DeeperGCN*	–	–	29.20±0.25	27.81±0.38
HIMP	–	78.80±0.82	–	–
PNA	85.19±0.99	79.05±1.32	–	–
DGN	84.70±0.47	79.70±0.97 –	–	
GINE*	–	–	30.65±0.30	29.17±0.15
PHC-GNN	82.17±0.89	79.34±1.16	30.68±0.25	29.47±0.26
Nested GIN*	83.17±1.99	78.34±1.86	29.15±0.35	28.32±0.41
Nested GIN* (ens)	80.80±2.78	**79.86**±1.05	30.59±0.56	**30.07**±0.37

- Check the paper to see more complete results

"Nested Graph Neural Networks, " Zhang & Li., NeurIPS 2021

Roadmap

- Distance features improve the expressive power of node representation refinement procedure [ZL, NeurIPS 2021]
  1. Definition for (general) distance features
  2. Standard GNNs miss capturing distance features
  3. How to use distance features in practice
     → node structural features, graph transformers
  4. How powerful distance features could be in theory

- Labeling tricks enable (most expressive) node representation refinement for node-set representation learning [ZLXWJ, NeurIPS 2021]

Labeling Trick: A Theory of Using Graph Neural Networks for Multi-Node Representation Learning, Zhang et al., NeurIPS 2021

Roadmap

- Distance features improve the expressive power of node representation refinement procedure [ZL, NeurIPS 2021]

  1. Definition for (general) distance features
  2. Standard GNNs miss capturing distance features
  3. How to use distance features in practice
     → node structural features, graph transformers
  4. How powerful distance features could be in theory

Distance encoding

[LWWL, NeurIPS 2020]

- Labeling tricks enable (most expressive) node representation refinement for node-set representation learning [ZLXWJ, NeurIPS 2021]

  1. Definition for node-set representation problems
  2. Which info does node representation refinement miss?
  3. How to use labeling tricks and how about their power?
  4. Comparison between distance features, labeling tricks and distance encoding.
  5. Comparison between SEAL and NBFNet.

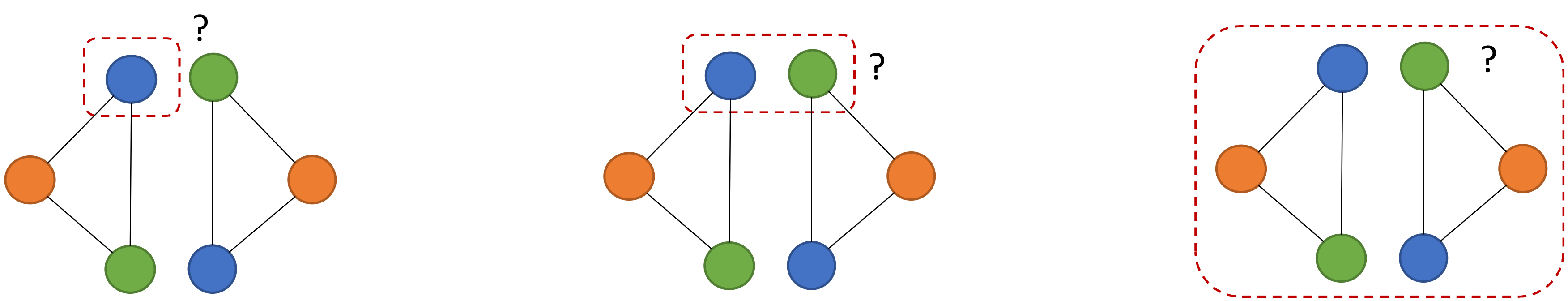
Labeling Trick: A Theory of Using Graph Neural Networks for Multi-Node Representation Learning, Zhang et al., NeurIPS 2021

Node-Set Representation Learning Problems

- Graph Data (A_i, X_i), $i = 1,2,\ldots$: the adjacency matrix A_i, possibly with node attributes X_i.

  --- One or several graphs do not matter.

- One Query (A_i, X_i, S_i): Make prediction for a set of nodes S_i.



$|S| = 1$ for node-level prediction

$|S| = 2$ for edge-level prediction

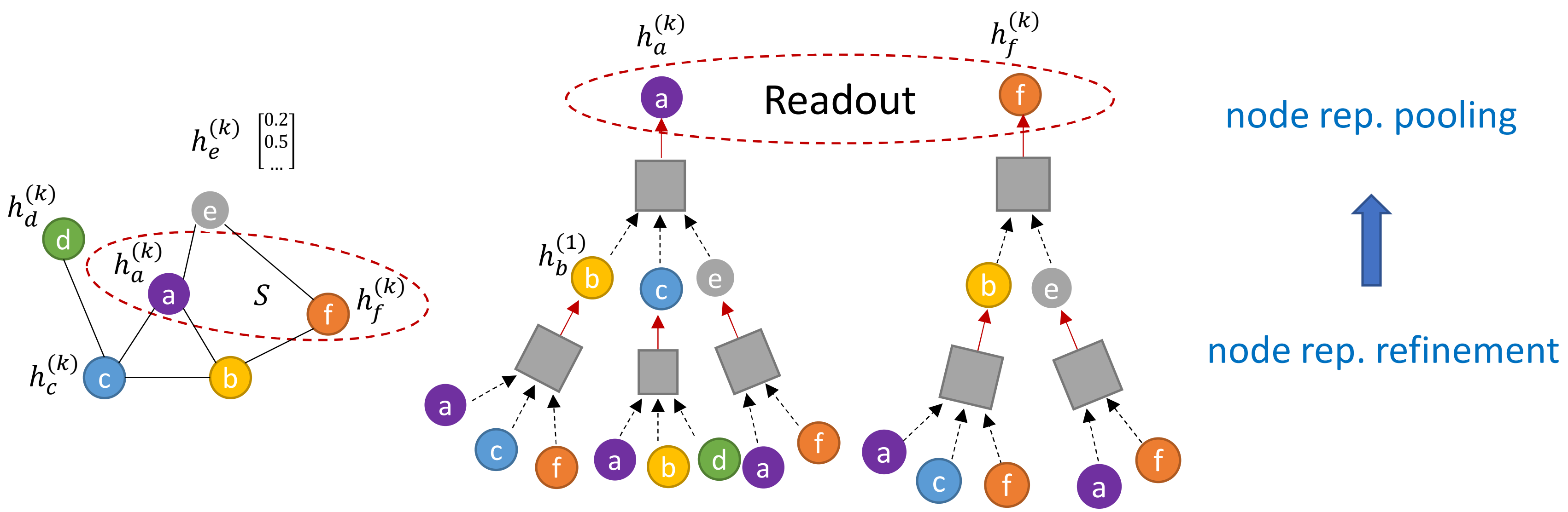
$|S| = |V|$ for graph-level prediction

Node-set representation learning

Crucial in predicting links, relations, network motifs

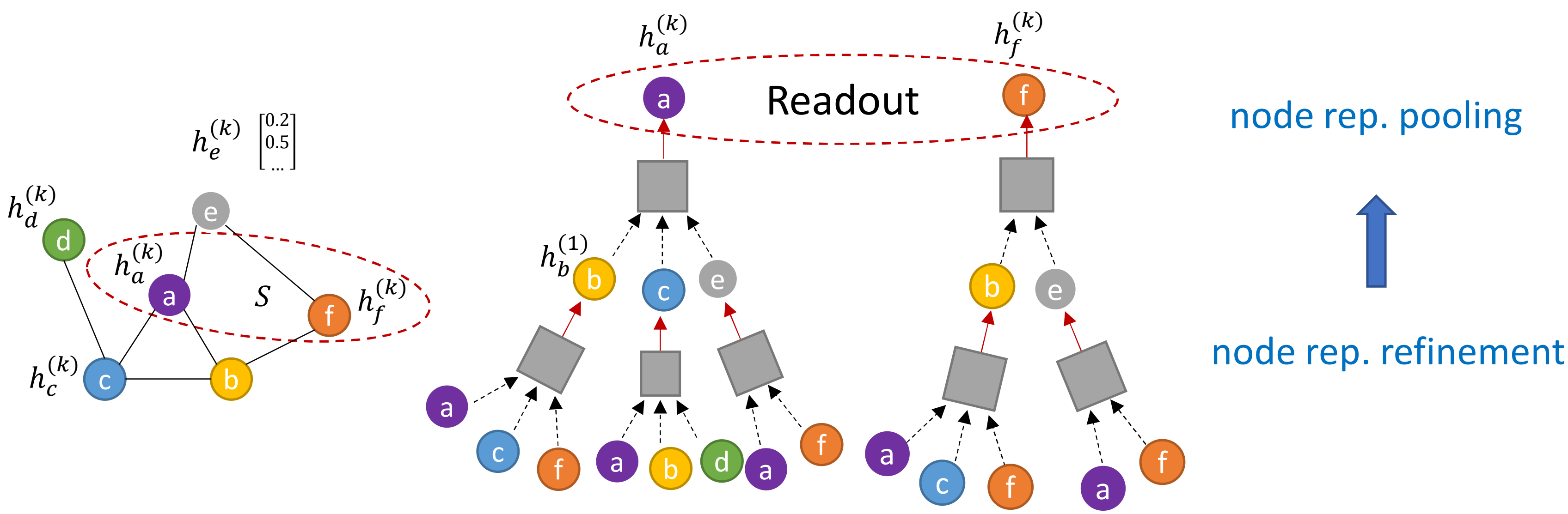
Node Representation Refinement Loses Information

- Most GNNs use (1) node rep. refinement + (2) node rep. pooling



Node Representation Refinement Loses Information

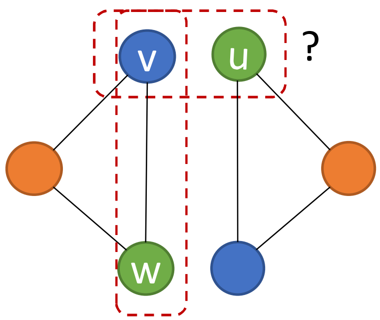
- Most GNNs use (1) node rep. refinement + (2) node rep. pooling
  - ➢ Node rep. refinement limits the expressive power to represent a node set, even if a more expressive GNN (thinking about using distance features) is used.



Node Representation Refinement Loses Information

- Most GNNs use (1) node rep. refinement + (2) node rep. pooling
  - Node rep. refinement limits the expressive power to represent a node set, even if a more expressive GNN (thinking about using distance features) is used.

- They fail to make the right prediction for node sets.

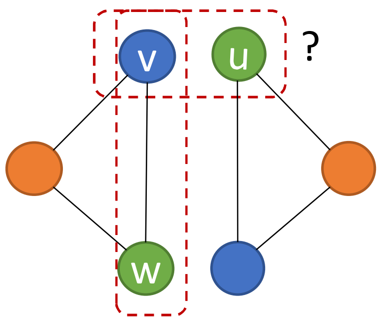


- Same-community query: Is node v more likely <u>in the same community with</u> node w or node u?

Node Representation Refinement Loses Information

- Most GNNs use (1) node rep. refinement + (2) node rep. pooling
  - ➢ Node rep. refinement limits the expressive power to represent a node set, even if a more expressive GNN (thinking about using distance features) is used.

- They fail to make the right prediction for node sets.



- Same-community query: Is node v more likely <u>in the same community with</u> node w or node u?

Most expressive GNNs fail: Node u and node w can be mapped to each other under graph automorphism

Position-aware Graph Neural Networks, You et al., ICML 2019

On the Equivalence between Positional Node Embeddings and Structural Graph Representations, Srinivasan & Ribeiro, ICLR 2020

Labeling Tricks

Labeling Tricks

For a graph query (A, X, S), a labeling trick gives extra node features $L \in R^{|V|}$ that satisfy two properties

(1) Distinguish S from the rest nodes.

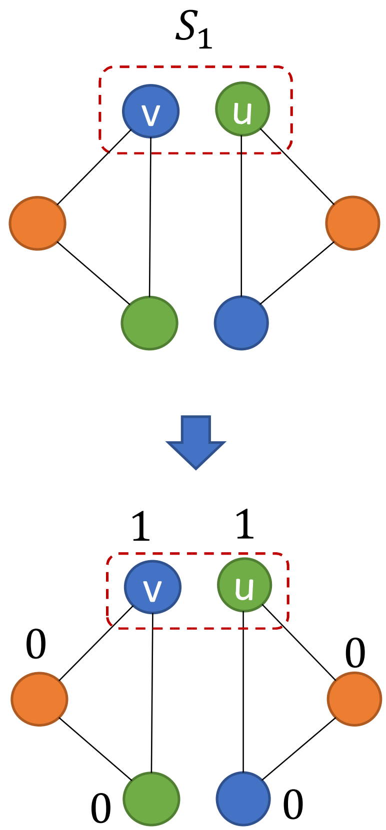
(2) Keep permutation equivariance. Later, we discuss this property more

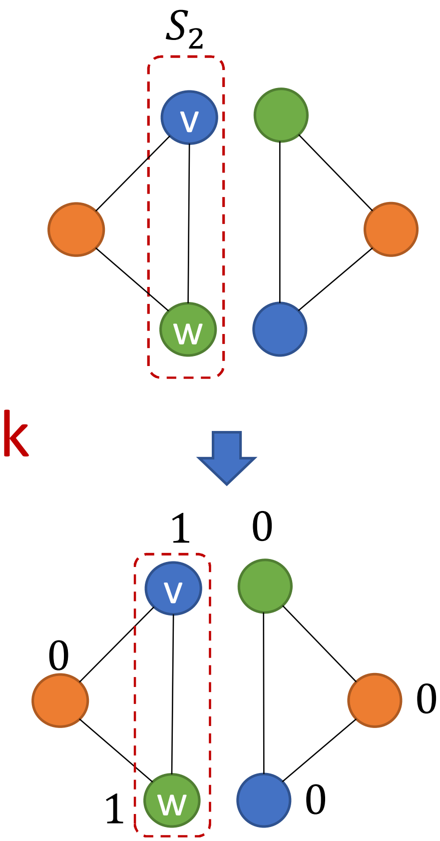
➢ [0-1 labeling trick] $L = 1_S$, i.e., a 0-1 vector with entries 1 for nodes in S and entries 0 for nodes out of S.

Labeling Trick: A Theory of Using Graph Neural Networks for Multi-Node Representation Learning, Zhang et al., NeurIPS 2021

Labeling Tricks

Q: Is node v more likely in the same community with node w or node u?





Plug in the 0-1 labeling trick

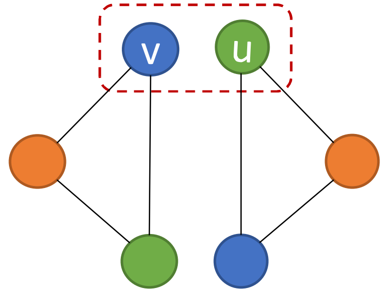
1-layer standard GNN can simply distinguish {v, u} and {v, w}.

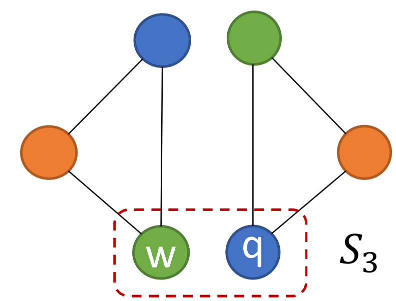
Labeling Trick: A Theory of Using Graph Neural Networks for Multi-Node Representation Learning, Zhang et al., NeurIPS 2021

Labeling Tricks

Q: Compare {v,u} with {w,q}?

S_1



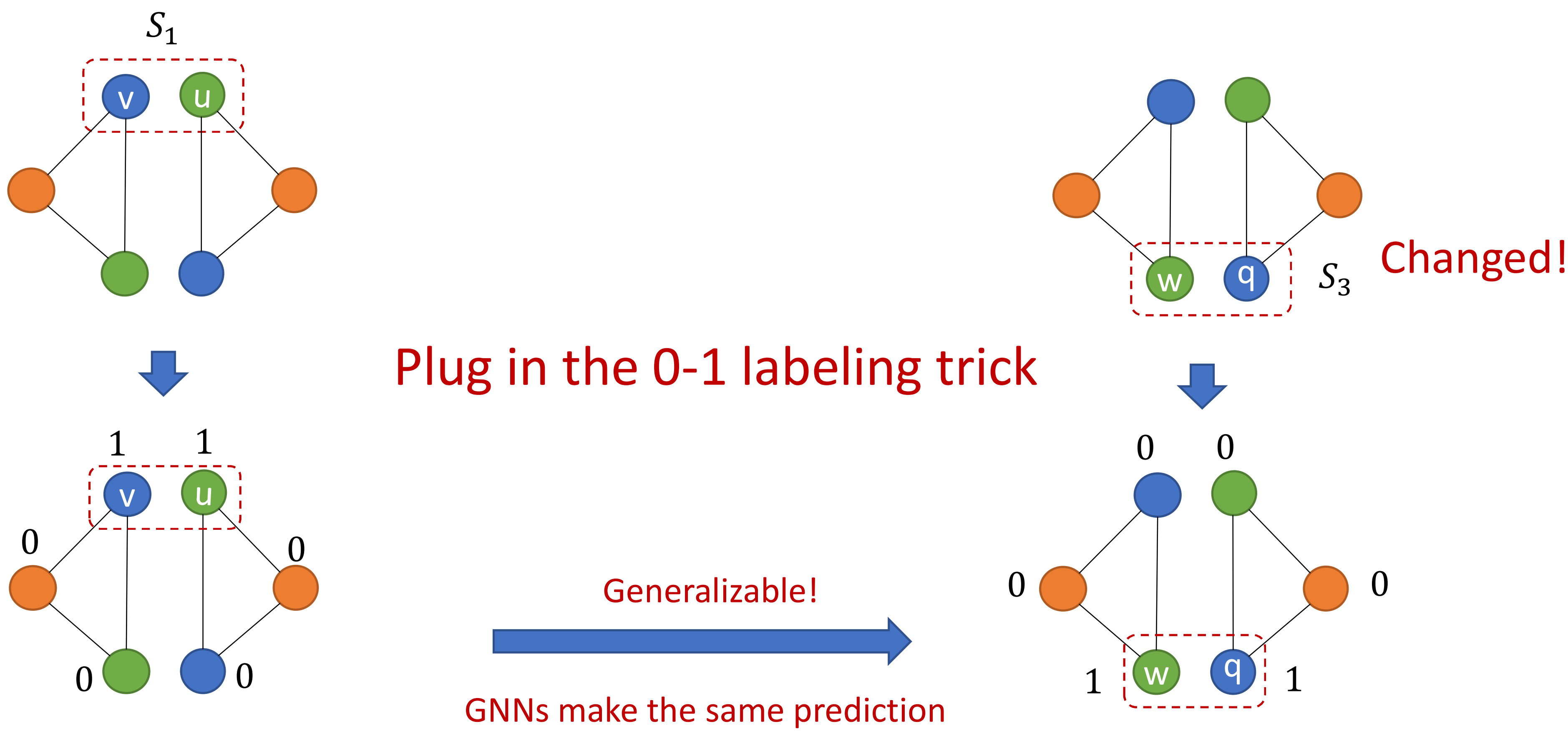


S_3 Changed!

Labeling Trick: A Theory of Using Graph Neural Networks for Multi-Node Representation Learning, Zhang et al., NeurIPS 2021

Labeling Tricks

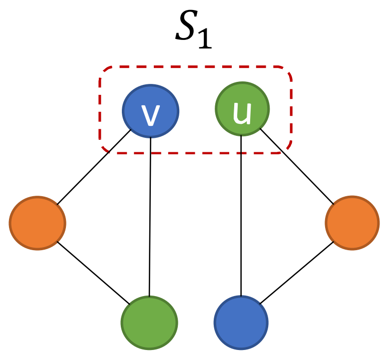
Q: Compare {v,u} with {w,q}?

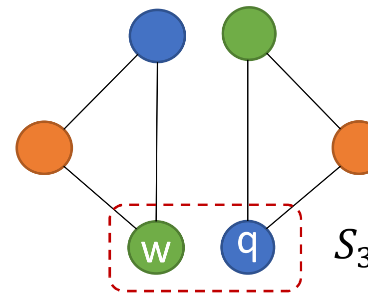


GNNs with labeling tricks can be generalized across
(1) different parts of the graph; (2) different graphs

Labeling Tricks

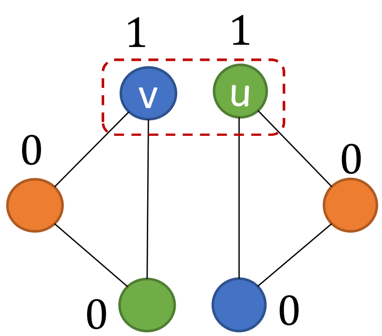
Q: Compare {v,u} with {w,q}?





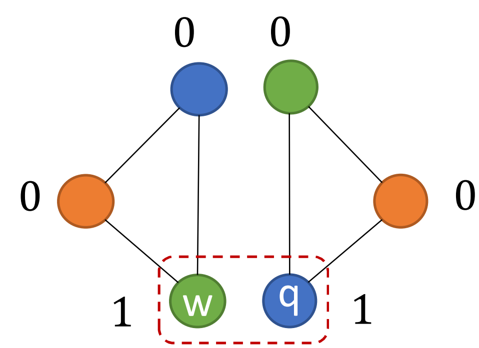
Changed!

Plug in the 0-1 labeling trick



Generalizable!

GNNs make the same prediction



Keep permutation equivariance → Inductive models

Labeling Tricks

Labeling Tricks

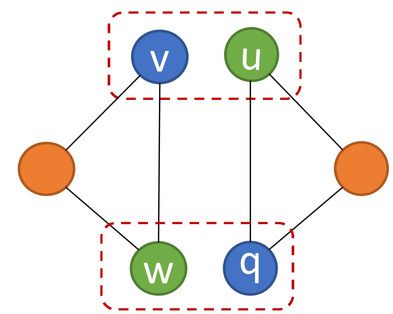
For a graph query (A, X, S), labeling trick gives extra node features $L \in R^{|V|}$ that satisfy two properties

(1) Distinguish S from the rest nodes.

(2) Keep permutation equivariance.

- Node-index-based encoding violates permutation equivariance

- Positional encoding (PE) based on Laplacian eigenmap violates permutation equivariance



$$(PE(v), PE(u)) \neq (PE(q), PE(w))$$

How to keep model inductive while using positional encoding? Check

Equivariant and Stable Positional Encoding for More Powerful Graph Neural Networks, Wang et al., ICLR 2022

How Powerful Labeling Tricks are?

Theorem [Zhang, Li, Xia, Wang, Jin, NeurIPS'21]

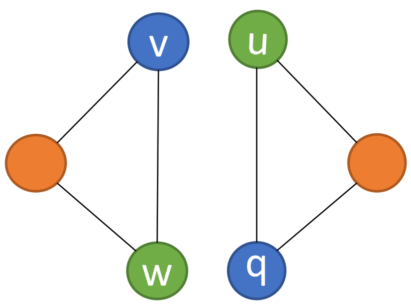
(Condition)

Suppose a GNN can distinguish non-isomorphic nodes, i.e., for two graphs (A, X) and (A', X') defined on the node set $V = [n]$, GNN gives two nodes the same representation

$[\cdot]_u$: uth row

$$[GNN(A, X)]_v = [GNN(A', X')]_u \quad ,$$

iff there is a permutation matrix P that $PAP^T = A'$, $PX = X'$ and $P_{uv} = 1$.



v, q are isomorphic $\quad [GNN(A, X)]_v = [GNN(A', X')]_q$

u, w are isomorphic $\quad [GNN(A, X)]_u = [GNN(A', X')]_w$

v, w are non-isomorphic $\quad [GNN(A, X)]_v \neq [GNN(A', X')]_w$

How Powerful Labeling Tricks are?

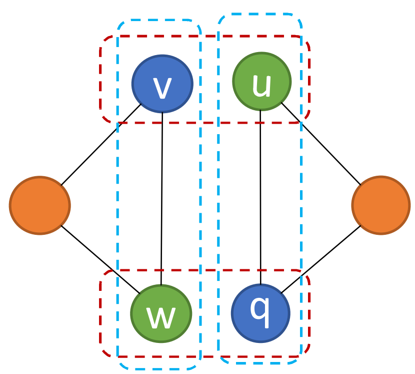
Theorem [Zhang, Li, Xia, Wang, Jin, NeurIPS'21]

(Conclusion)

For any two node-set queries (A, X, S) and (A', X', S'), the above GNN + set pooling paired with labeling tricks L, L' (satisfying equivariance) can distinguish two non-isomorphic queries, i.e.,

$$\{[GNN(A, X \oplus L)]_v \ | v \in S\} = \{[GNN(A', X' \oplus L')]_u \ | u \in S'\}$$

iff there is a permutation matrix P that $PAP^T = A'$, $PX = X'$ and $P_{SS'}$ is a sub permutation matrix.



$\{v, u\}$ and $\{w, q\}$ are isomorphic
$\{v, w\}$ and $\{u, q\}$ are isomorphic
but $\{v, u\}$ and $\{v, w\}$ are non-isomorphic

With labeling tricks, we can distinguish $\{v, u\}$ and $\{v, w\}$

Labeling tricks fill in the gap between node representation refinement and the prediction over node-sets

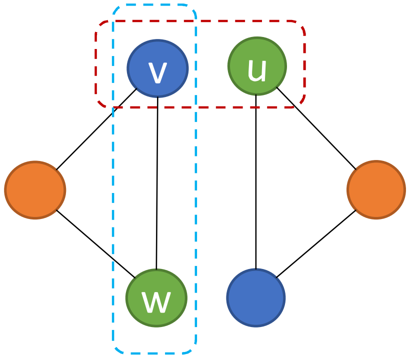
How Powerful Labeling Tricks are?

- Is the requirement on distinguishing non-isomorphic nodes too strong to make labeling tricks trivial?

- [Informal Theorem 2] No! Many node pairs $\{v, u\}$ and $\{v, w\}$:

  → GNNs that distinguish non-isomorphic nodes cannot distinguish these node pairs

  → Standard GNNs plus labeling tricks can distinguish these node pairs



Labeling Trick: A Theory of Using Graph Neural Networks for Multi-Node Representation Learning, Zhang et al., NeurIPS 2021

Distance Features, Labeling Tricks and Distance Encoding

Any questions?

Distance Features, Labeling Tricks and Distance Encoding

- Distance features do not depend on the queried node set S.

  → A general feature to improve the expressive power of the node representation refinement procedure.

Distance features between two nodes

Given a graph A, the distance from node u to node v is $\zeta(v|A,u) = \left((W)_{uv}, (W^2)_{uv}, \ldots, \left(W^k\right)_{uv}\right)$. W is the random walk matrix.

Distance Features, Labeling Tricks and Distance Encoding

- Distance features do not depend on the queried node set S.

  → A general feature to improve the expressive power of the node representation refinement procedure.

- Labeling tricks specifically depend on the queried node set S.

  → Distinguish the nodes in the queried node set from the rest nodes to improve the expressive power of the representation of a queried node set.

Labeling Trick

For a graph query (A, X, S), labeling trick gives extra node features $L \in R^{|V|}$ that satisfy two properties

(1) Distinguish S from the rest nodes.

(2) Keep permutation equivariance.

Distance Features, Labeling Tricks and Distance Encoding

- Distance features do not depend on the queried node set S.

  → A general feature to improve the expressive power of the node representation refinement procedure.

- Labeling tricks specifically depend on the queried node set S.

  → Distinguish the nodes in the queried node set from the rest nodes to improve the expressive power of the representation of a queried node set.

- Distance encoding mixes these two concepts

  → Use the distance features to the queried node set S as a labeling trick

Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020

Distance Features, Labeling Tricks and Distance Encoding

- Distance features do not depend on the queried node set S.

  → A general feature to improve the expressive power of the node representation refinement procedure.

- Labeling tricks specifically depend on the queried node set S.

  → Distinguish the nodes in the queried node set from the rest nodes to improve the expressive power of the representation of a queried node set.

- Distance encoding mixes these two concepts

  → Use the distance features to the queried node set S as a labeling trick

Distance Encoding

Given a query (A, X, S), encode each node v with

$$\zeta(v|A,S) = \{\zeta(v|A,u)|u \in S\},$$

where $\zeta(v|A,u)$ is the distance feature between v and u.

Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020

Distance Encoding and 0-1 Labeling Trick

- Distance encoding satisfies the two properties of labeling tricks
  - → Distinguish the queried node set
  - → Keep permutation equivariance

Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020

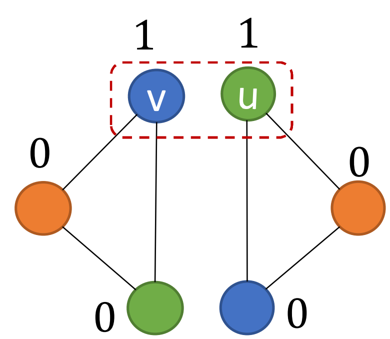
Distance Encoding and 0-1 Labeling Trick

- ➢ Distance encoding satisfies the two properties of labeling tricks

  → Distinguish the queried node set
  → Keep permutation equivariance

- ➢ 0-1 labeling trick is a special case of distance encoding

  → "node v is associated with 1 in 0-1 labeling trick" means "the shortest path distance from node v to S is 0"



Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020

Distance Encoding and 0-1 Labeling Trick

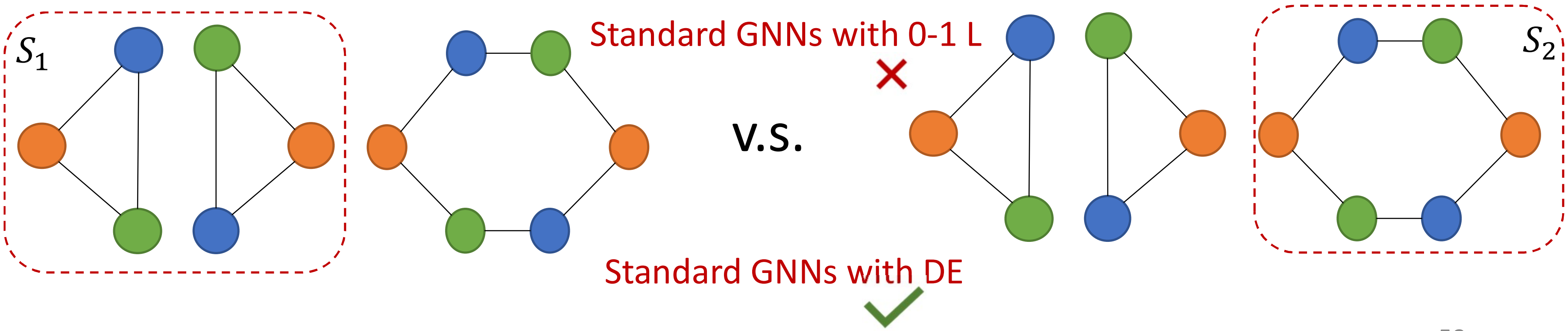
- Distance encoding satisfies the two properties of labeling tricks

  → Distinguish the queried node set
  → Keep permutation equivariance

- 0-1 labeling trick is a special case of distance encoding

  → "node v is associated with 1 in 0-1 labeling trick" means "the shortest path distance from node v to S is 0"

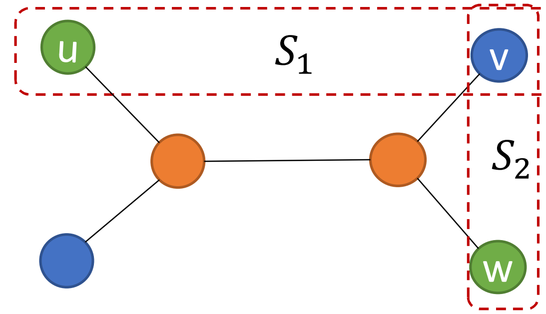
- GNNs with 0-1 labeling trick could be less expressive than GNNs with distance encoding



Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020

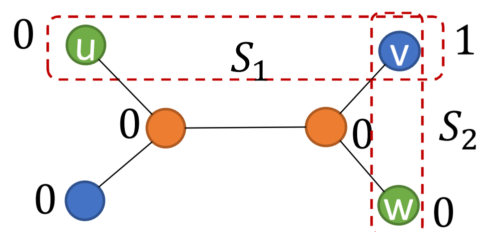
Labeling Trick for Link Prediction --- SEAL v.s. NBFNet

Q: Is node v more likely linked to node w or node u?



- SEAL: label two queries respectively

- NBFNet (or IDGNN): label a source node overlapped by two queries



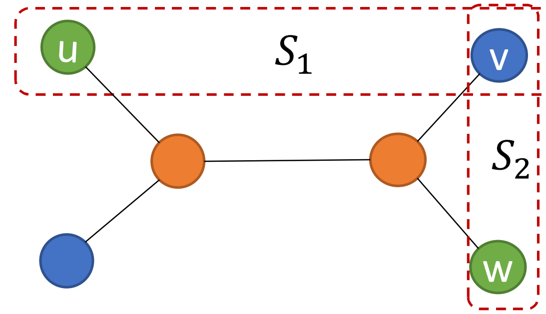
SEAL: “Link Prediction Based on Graph Neural Networks, ” Zhang & Chen, NeurIPS 2018

IDGNN: “Identity-aware Graph Neural Networks, ” You et al., AAAI 2021

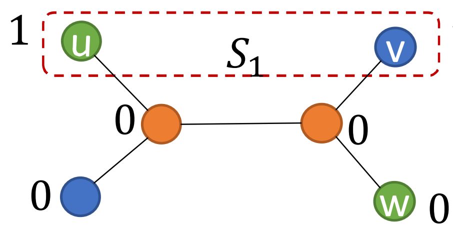
NBFNet: “Neural Bellman-Ford Networks: A General Graph Neural Network Framework for Link Prediction, ” Zhu et al., NeurIPS 2021

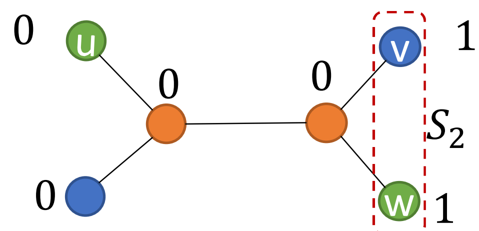
Labeling Trick for Link Prediction --- SEAL v.s. NBFNet

Q: Is node v more likely linked to node w or node u?



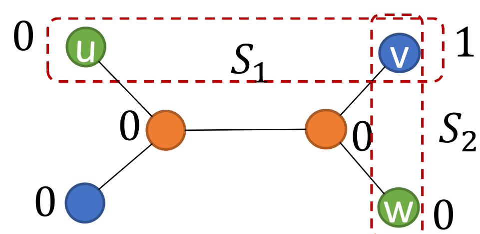
- SEAL: label two queries respectively





- NBFNet (or IDGNN): label a source node overlapped by two queries

Run one-time GNN and Readout {u,v} and {w,v} for prediction; Save computation



SEAL: “Link Prediction Based on Graph Neural Networks, ” Zhang & Chen, NeurIPS 2018

IDGNN: “Identity-aware Graph Neural Networks, ” You et al., AAAI 2021

NBFNet: “Neural Bellman-Ford Networks: A General Graph Neural Network Framework for Link Prediction, ” Zhu et al., NeurIPS 2021

SEAL is theoretically more powerful than NBFNet

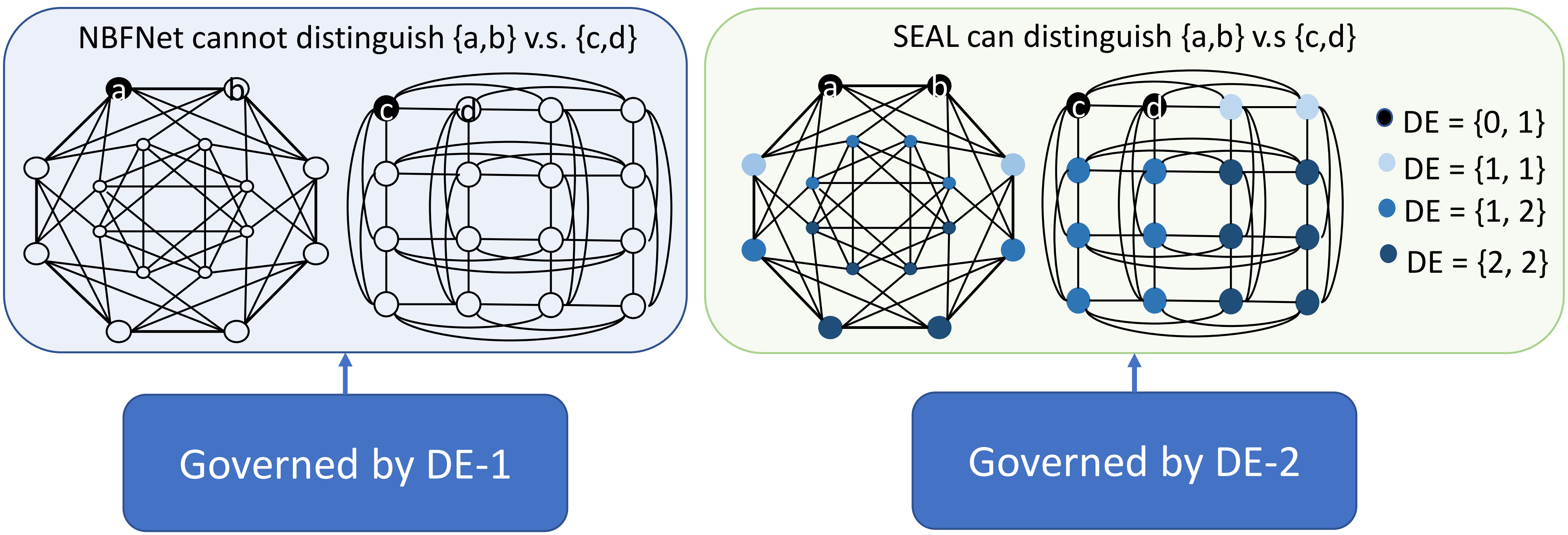
- The theory behind distance encoding tells the difference.
  - The expressive power of SEAL is governed by DE-2
  - The expressive power of NBFNet is governed by DE-1

DE-k, where k is the size of node set used to compute distance encoding --- [Li et al., NeurIPS 2020]

Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020

SEAL is theoretically more powerful than NBFNet

- The theory behind distance encoding tells the difference.
  - The expressive power of SEAL is governed by DE-2
  - The expressive power of NBFNet is governed by DE-1
- Task: Distinguish links in two strongly regular graphs (distance regular graphs more precisely)



Distance encoding: Design provably more powerful neural networks for graph representation learning, Li et al., NeurIPS 2020
The Expressive Power of Graph Neural Networks, Li & Leskovec, Book Chapter 5, Figure 5.14, 2021

Empirical Results

Table 1: Results for ogbl-ppa, ogbl-collab, ogbl-ddi and ogbl-citation2.

		ogbl-ppa Hits@100 (%)		ogbl-collab Hits@50 (%)		ogbl-ddi Hits@20 (%)		ogbl-citation2 MRR (%)	
Category	**Method**	Validation	**Test**	Validation	**Test**	Validation	**Test**	Validation	**Test**
Non-GNN	**CN**	28.23±0.00	27.6±0.00	60.36±0.00	61.37±0.00	9.47±0.00	17.73±0.00	51.19±0.00	51.47±0.00
	AA	32.68±0.00	32.45±0.00	63.49±0.00	64.17±0.00	9.66±0.00	18.61±0.00	51.67±0.00	51.89±0.00
	MLP	0.46±0.00	0.46±0.00	24.02±1.45	19.27±1.29	–	–	29.03±0.17	29.06±0.16
	Node2vec	22.53±0.88	22.26±0.88	57.03±0.52	48.88±0.54	32.92±1.21	23.26±2.09	61.24±0.11	61.41±0.11
	MF	32.28±4.28	32.29±0.94	48.96±0.29	38.86±0.29	33.70±2.64	13.68±4.75	51.81±4.36	51.86±4.43
Plain GAE	**GraphSAGE**	17.24±2.64	16.55±2.40	56.88±0.77	54.63±1.12	62.62±0.37	53.90±4.74	82.63±0.23	82.60±0.36
	GCN	18.45±1.40	18.67±1.32	52.63±1.15	47.14±1.45	55.50±2.08	37.07±5.07	84.79±0.23	84.74±0.21
	GCN+LRGA	25.75±2.82	26.12±2.35	60.88±0.59	52.21±0.72	**66.75**±0.58	**62.30**±9.12	66.48±1.61	66.49±1.59
Labeling Trick	**GCN+DE**	36.31±3.59	36.48±3.78	64.13±0.16	64.44±0.29	29.85±2.25	26.63±6.82	60.17±0.63	60.30±0.61
	GCN+DRNL	46.43±3.03	45.24±3.95	64.51±0.42	64.40±0.45	29.47±1.54	22.81±4.93	81.07±0.30	81.27±0.31
	SEAL	**51.25**±2.52	**48.80**±3.16	**64.95**±0.43	**64.74**±0.43	28.49±2.69	30.56±3.86	**87.57**±0.31	**87.67**±0.32

- Labeling trick-based GNNs are generally better than baselines
- The DDI dataset:
  - Baselines use node-index-based encoding.
  - The generalization issue behind node-index-based encoding is reduced because # of edges is large and # of nodes is small.

Roadmap

- Distance features improve the expressive power of node representation refinement procedure [ZL, NeurIPS 2021]
  1. Definition for (general) distance features
  2. Standard GNNs miss capturing distance features
  3. How to use distance features in practice
     → node structural features, graph transformers
  4. How powerful distance features could be in theory

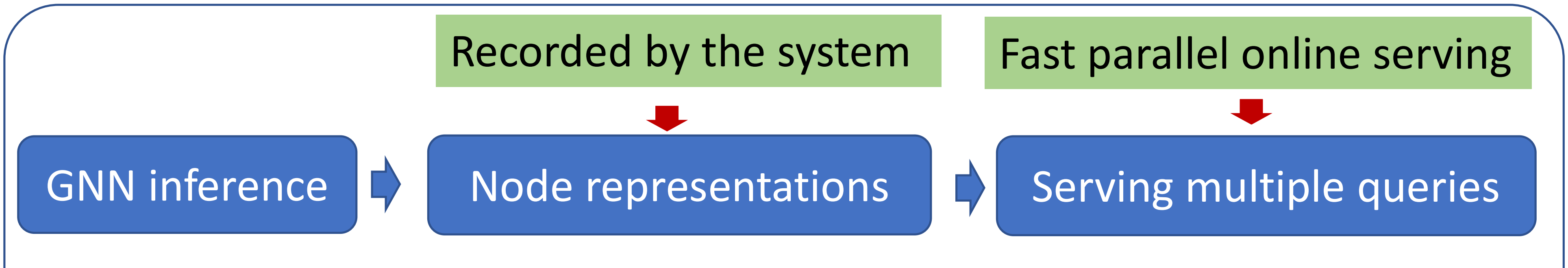
Distance encoding
[LWWL, NeurIPS 2020],…

- Labeling tricks complement fundamental drawbacks of (even most expressive) node representation refinement [ZLXWJ, NeurIPS 2021]
  1. Definition for node-set representation problems
  2. Which info does node representation refinement miss
  3. How to use labeling tricks and how about their power?
  4. Comparison between distance features, labeling tricks and distance encoding.
  5. Comparison between SEAL and NBFNet.

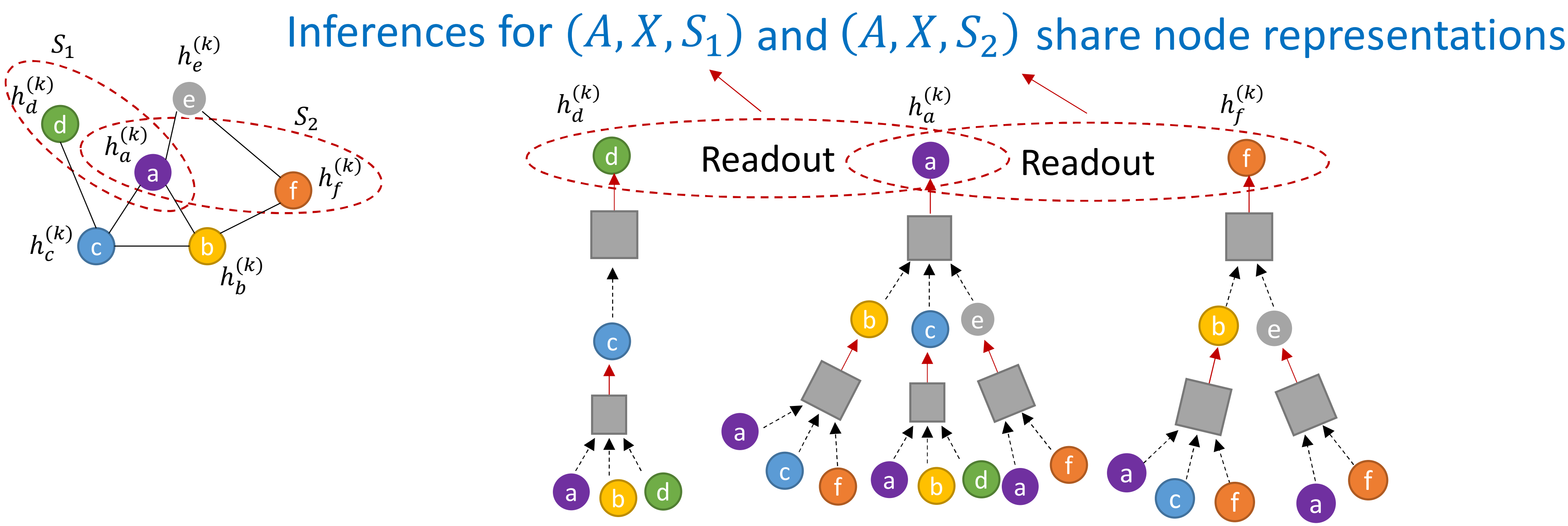
- **Scalable Distance Encoding** [YZWWL, submitted]

Algorithm and System Co-design for Efficient Subgraph-based Graph Representation Learning. Yin et al., submitted

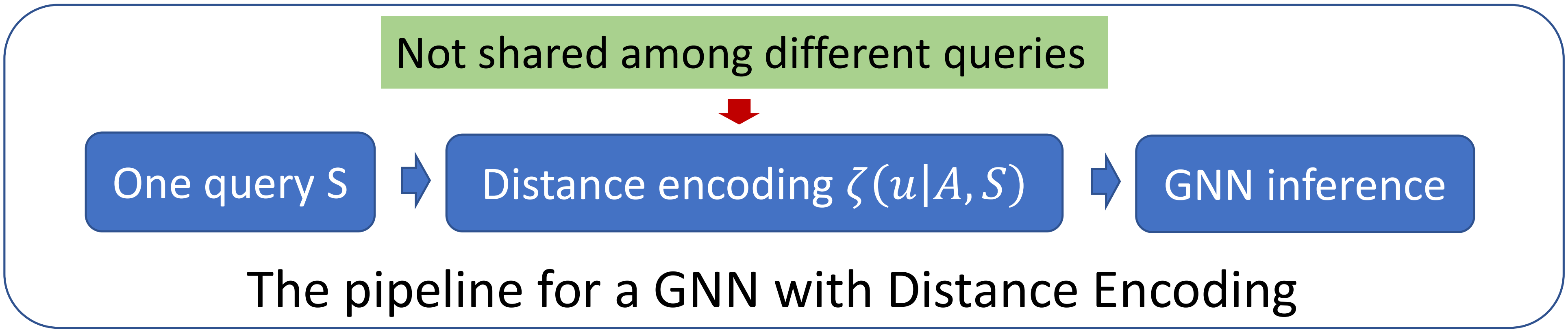
Computation Challenge of Distance Encoding



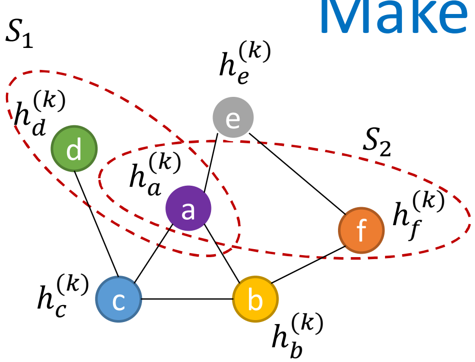
The pipeline for a standard GNN

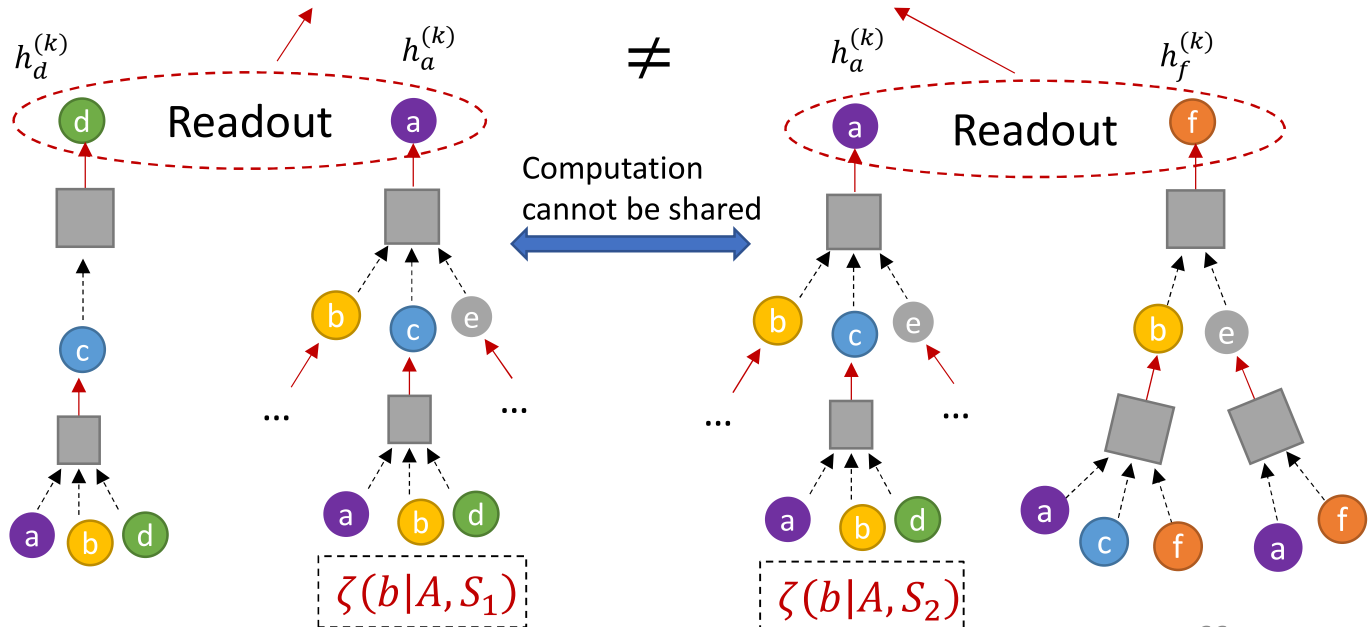


Computation Challenge of Distance Encoding



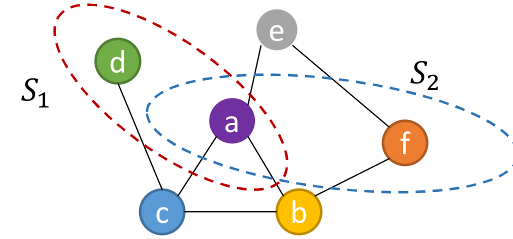
Make inference (A, X, S_1) and (A, X, S_2) separately

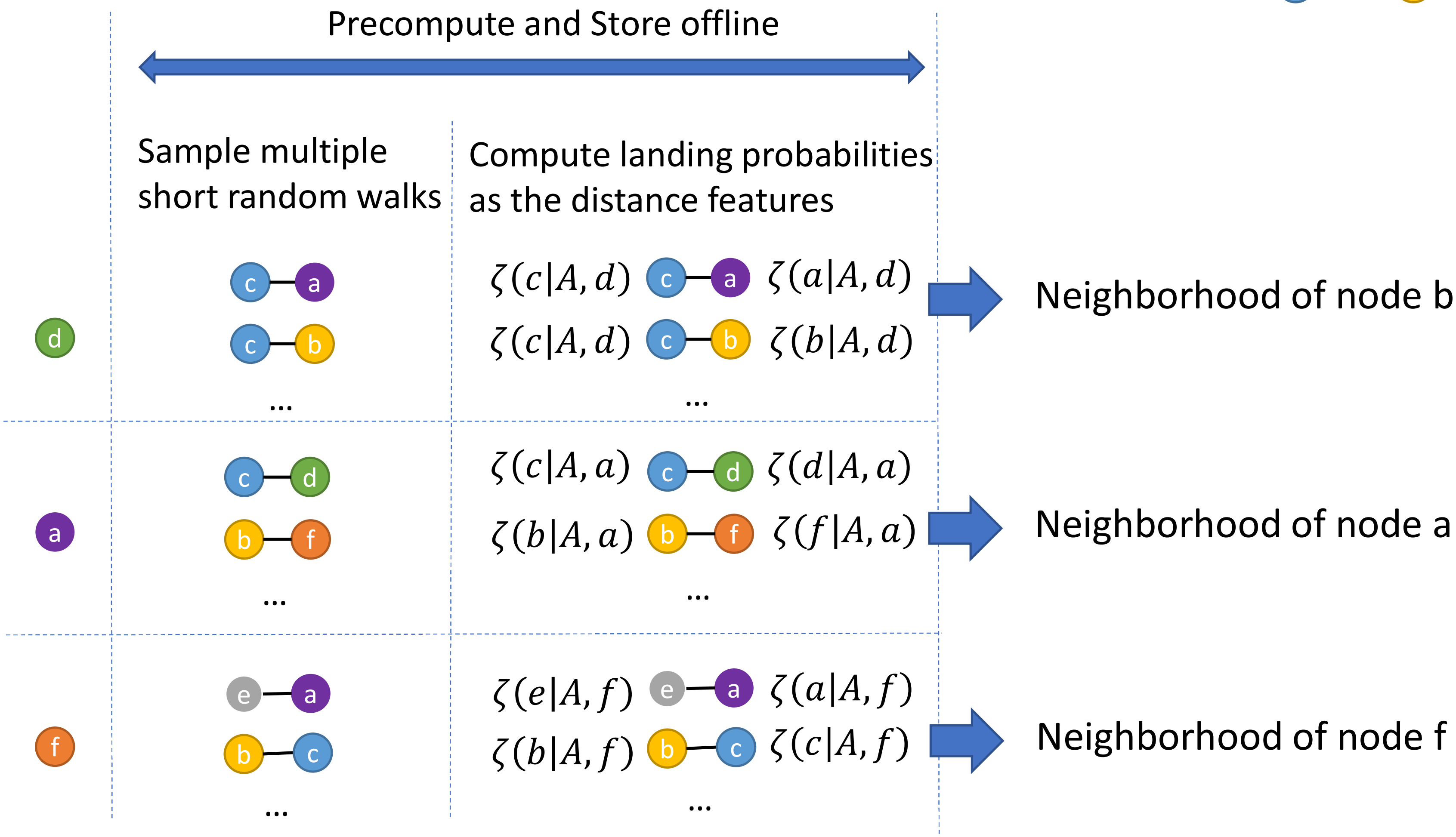




Our Solution

- Key idea: Make distance features precomputed and shared as much as possible.

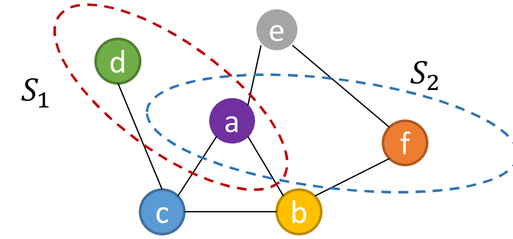


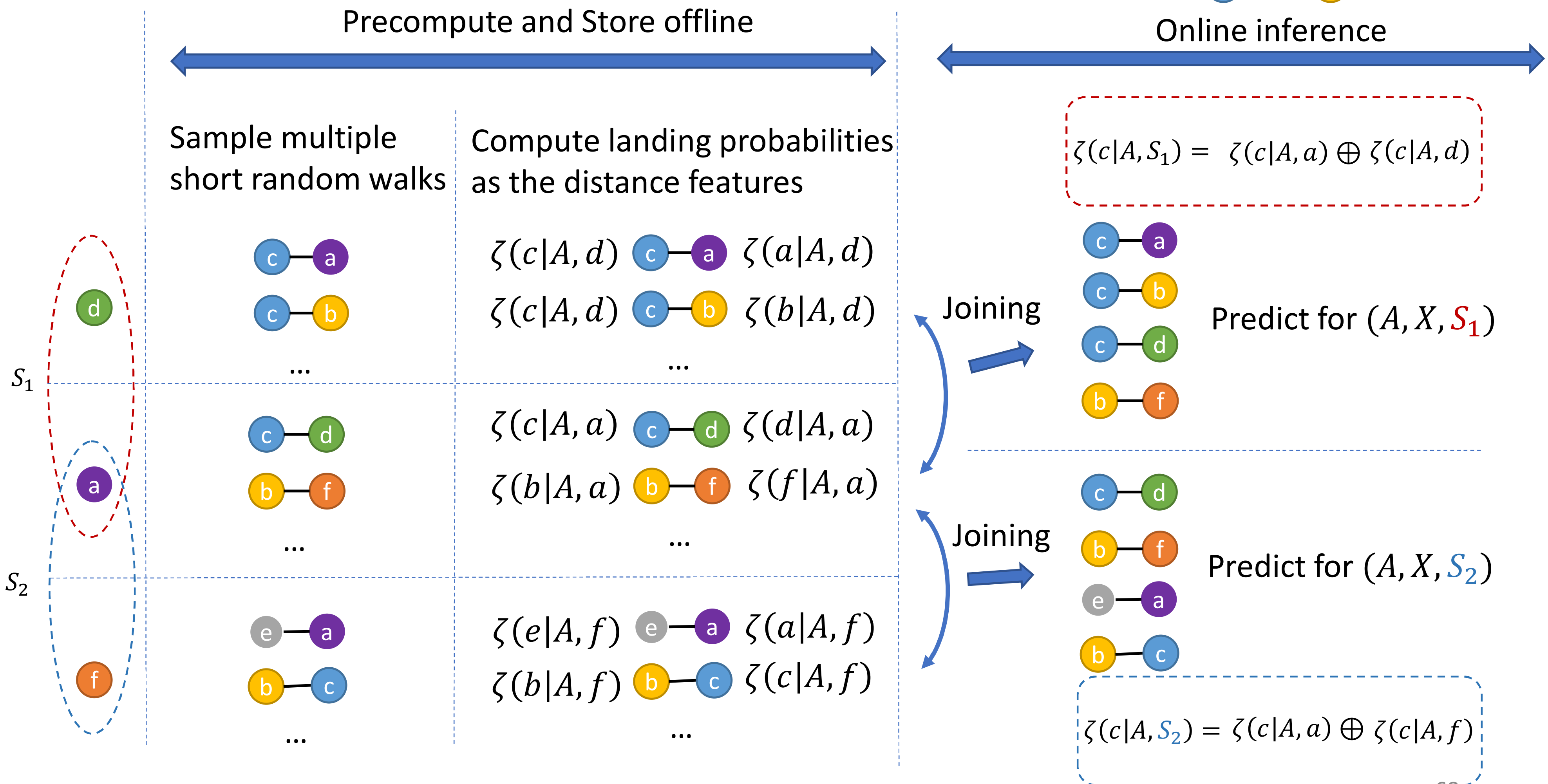


Algorithm and System Co-design for Efficient Subgraph-based Graph Representation Learning. Yin et al., submitted

Our Solution

- Key idea: Make distance features precomputed and shared as much as possible.





Algorithm and System Co-design for Efficient Subgraph-based Graph Representation Learning. Yin et al., submitted

Empirical Results

Table 3: Results for Link Prediction on OGB.

Models	citation2 MRR (%)	collab Hits@50 (%)	ppa Hits@100 (%)
Node2vec	61.28±0.15	47.54±0.78	18.05±0.52
DeepWalk	84.47±0.04	49.08±0.93	27.80±1.71
Marius	72.53±0.14	19.31±1.01	31.24±2.28
GCN	84.74±0.21	44.75±1.07	18.67±1.32
SAGE	82.60±0.36	54.63±1.12	16.55±2.40
Cluster-GCN	80.04±0.25	44.02±1.37	3.56±0.40
GraphSAINT	79.85±0.40	53.12±0.52	3.83±1.33
SEAL	87.67±0.32	**63.64±0.71**	48.80±3.16
SUREL	**89.74±0.18**	63.34±0.52	**53.23±1.03**

Table 5: Breakdown of the Runtime, Memory Consumption for Different Models on citation2, collab, and DBLP-coauthor. Training time is calculated if no better validation performance is observed in 3 consecutive epochs, which assumes the model has converged.

	Models	Runtime (s)				Memory (GB)	
		Prep.	Train	Inf.	Total	RAM	GPU
citataion2	**GCN**	17	16,835	32	16,884	9.5	37.55
	Cluster-GCN	197	2,663	82	2,942	18.3	14.07
	GraphSAINT	140	3,845	86	4,071	16.9	14.77
	SEAL (1-hop)	46	22,296	130,312	152,654	36.5	3.35
	SUREL	31	2,096	7,959	10,086	15.2	4.50
collab	**GCN**	6	840	0.1	846	3.2	5.17
	Cluster-GCN	8	649	0.2	666	3.4	5.29
	GraphSAINT	<1	6,746	0.2	6,747	3.2	6.58
	SEAL (1-hop)	10	7,675	37	7,722	15.4	6.97
	SUREL	<1	1,720	8	1,728	3.6	5.57

- Comparable or even better prediction accuracy than SEAL on link prediction tasks

- Training time comparable with standard GNNs
- Inference time 5~20 times faster than SEAL though still much slower than standard GNNs

Summary

- Distance features improve the expressive power of node representation refinement procedure [ZL, NeurIPS 2021]
- Labeling tricks enable (most expressive) node representation refinement for node-set representation learning [ZLXWJ, NeurIPS 2021]
- Distance encoding is a distance-feature-based labeling trick. [LWWL, NeurIPS 2020]
- Scalable Distance Encoding [YZWWL, submitted]

Thanks!

Backup

Empirical Results

Table 1: Results for ogbl-ppa, ogbl-collab, ogbl-ddi and ogbl-citation2.

		ogbl-ppa Hits@100 (%)		ogbl-collab Hits@50 (%)		ogbl-ddi Hits@20 (%)		ogbl-citation2 MRR (%)	
Category	**Method**	Validation	**Test**	Validation	**Test**	Validation	**Test**	Validation	**Test**
Non-GNN	**CN**	28.23±0.00	27.6±0.00	60.36±0.00	61.37±0.00	9.47±0.00	17.73±0.00	51.19±0.00	51.47±0.00
	AA	32.68±0.00	32.45±0.00	63.49±0.00	64.17±0.00	9.66±0.00	18.61±0.00	51.67±0.00	51.89±0.00
	MLP	0.46±0.00	0.46±0.00	24.02±1.45	19.27±1.29	–	–	29.03±0.17	29.06±0.16
	Node2vec	22.53±0.88	22.26±0.88	57.03±0.52	48.88±0.54	32.92±1.21	23.26±2.09	61.24±0.11	61.41±0.11
	MF	32.28±4.28	32.29±0.94	48.96±0.29	38.86±0.29	33.70±2.64	13.68±4.75	51.81±4.36	51.86±4.43
Plain GAE	**GraphSAGE**	17.24±2.64	16.55±2.40	56.88±0.77	54.63±1.12	62.62±0.37	53.90±4.74	82.63±0.23	82.60±0.36
	GCN	18.45±1.40	18.67±1.32	52.63±1.15	47.14±1.45	55.50±2.08	37.07±5.07	84.79±0.23	84.74±0.21
	GCN+LRGA	25.75±2.82	26.12±2.35	60.88±0.59	52.21±0.72	**66.75**±0.58	**62.30**±9.12	66.48±1.61	66.49±1.59
Labeling Trick	**GCN+DE**	36.31±3.59	36.48±3.78	64.13±0.16	64.44±0.29	29.85±2.25	26.63±6.82	60.17±0.63	60.30±0.61
	GCN+DRNL	46.43±3.03	45.24±3.95	64.51±0.42	64.40±0.45	29.47±1.54	22.81±4.93	81.07±0.30	81.27±0.31
	SEAL	**51.25**±2.52	**48.80**±3.16	**64.95**±0.43	**64.74**±0.43	28.49±2.69	30.56±3.86	**87.57**±0.31	**87.67**±0.32

- Labeling trick-based GNNs are generally better than baselines
- The DDI dataset:
  - Baselines use node-index-based encoding.
  - The generalization issue behind node-index-based encoding is reduced because # of edges is large and # of nodes is small.
- NBFNet suffers from overflow issues here (more engineering works are needed)