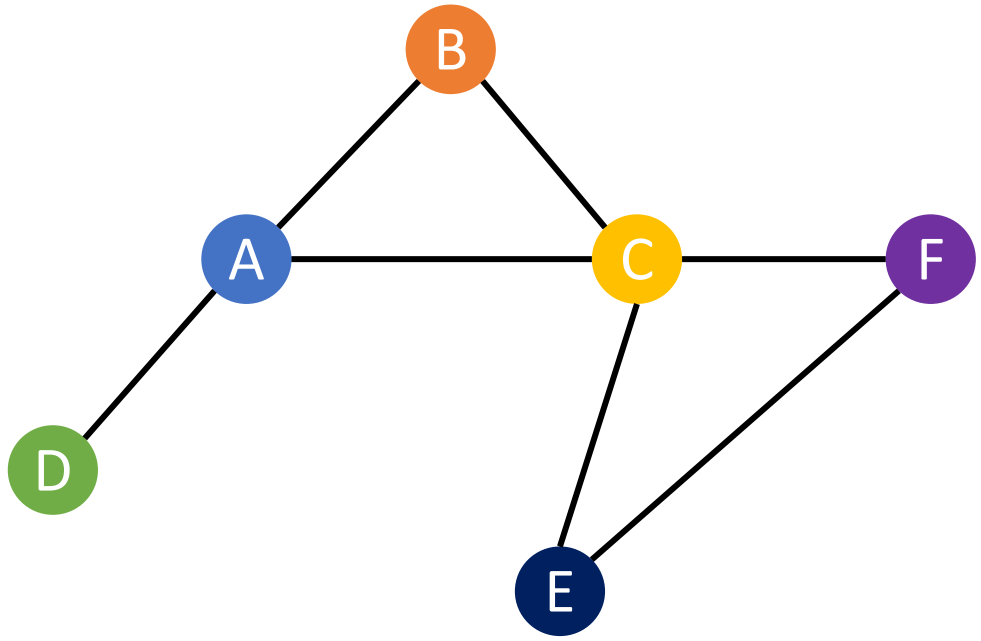
Introduction to Graph Deep Learning

Minji Yoon
Computer Science Department
Carnegie Mellon University

Talk objectives

- Introduce Graph Neural Networks (GNNs)
- Highlight interesting open research questions

What is a graph?

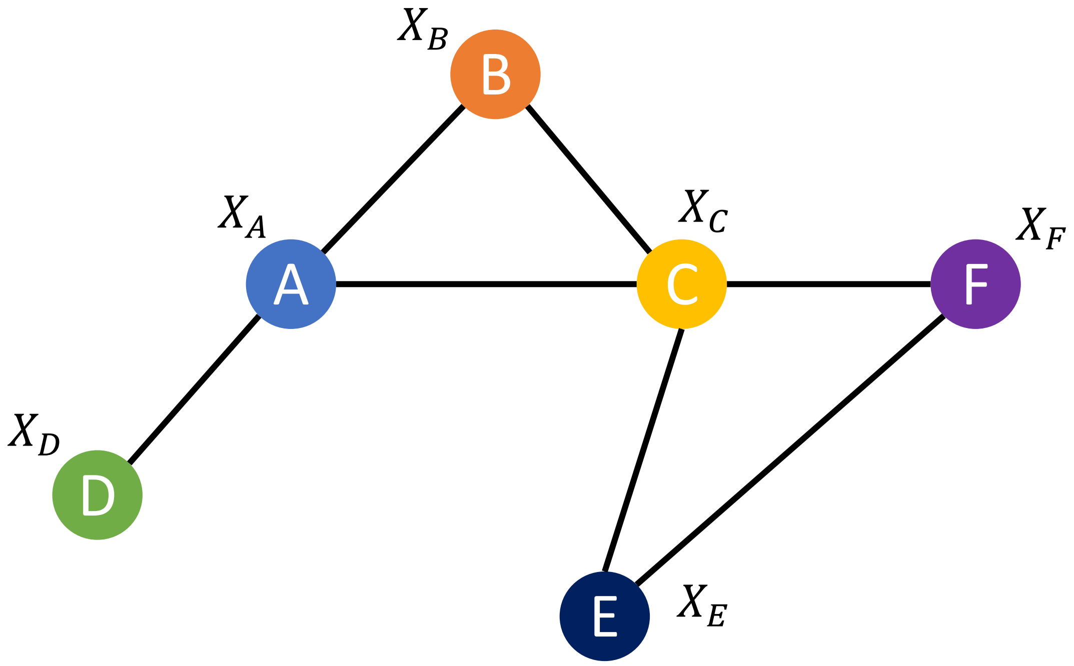


A graph is composed of
- **Nodes** (also called vertices)
- **Edges** connecting a pair of nodes

presented in an **adjacency matrix**

	A	B	C	D	E	F
A		1	1	1		
B	1		1			
C	1	1			1	1
D	1					
E			1			1
F			1		1	

What is a graph?



A graph is composed of

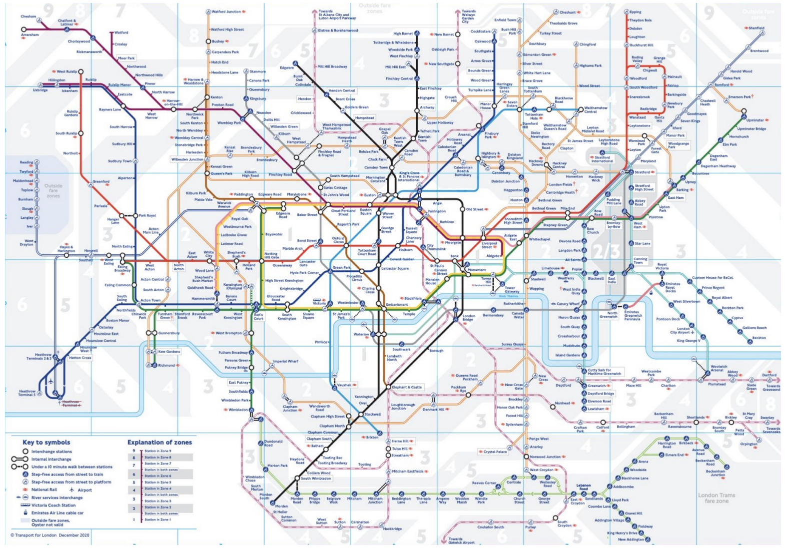
- **Nodes** (also called vertices)
- **Edges** connecting a pair of nodes

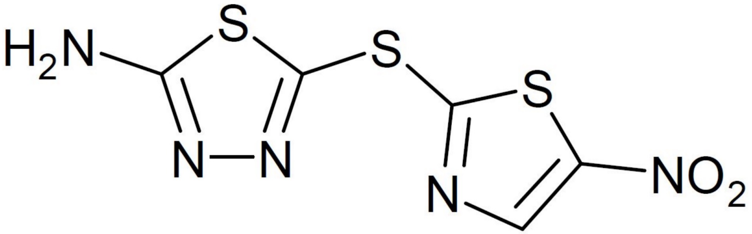
presented in an **adjacency matrix**

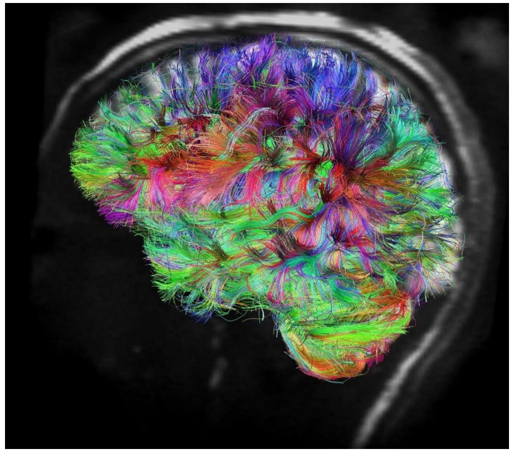
Nodes can have **feature vectors**

A	X_A
B	X_B
C	X_C
D	X_D
E	X_E
F	X_F

Graphs are everywhere







Graph Neural Networks have a large impact on…

DeepMind > Blog > Traffic prediction with advanced Graph Neural Networks

BLOG POST
RESEARCH
03 SEP 2020

Traffic prediction with advanced Graph Neural Networks

Pinterest Engineering
Aug 15, 2018 · 8 min read

PinSage: A new graph convolutional neural network for web-scale recommender systems

Ruining He | Pinterest engineer, Pinterest Labs

Web image search gets better with graph neural networks

ı to image search uses images returned by traditional search
des in a graph neural network through which similarity signals are
ıieving improved ranking in cross-modal retrieval.

ıral Network
'ER LABS Europe NAVER LABS FOLLOW

amazon | science

PUBLICATION

P-Companion: A principled framework for diversified complementary product recommendation

By Junheng Hao, Tong Zhao, Jin Li, Xin Luna Dong, Christos Faloutsos, Yizhou Sun, Wei Wang
2020

Food Discovery with Uber Eats: Using Graph Learning to Power Recommendations

Ankit Jain, Isaac Liu, Ankur Sarda, and Piero Molino December 4, 2019
0

Graph Neural Networks have a large impact on...

GCN-RL Circuit Designer: Transferable Transistor Sizing with Graph Neural Networks and Reinforcement Learning

Hanrui Wang[1], Kuan Wang[1], Jiacheng Yang[1], Linxiao Shen[2], Nan Sun[2], Hae-Seung Lee[1], Song Han[1]

[1]Massachusetts Institute of Technology

[2]UT Austin

HAN LAB Hardware, AI and Neural-nets

TEXAS The University of Texas at Austin

npj | computational materials

Explore content ⌄ About the journal ⌄ Publish with us ⌄

nature > npj computational materials > articles > article

Article | Open Access | Published: 03 June 2021

Benchmarking graph neural networks for materials chemistry

Victor Fung, Jiaxin Zhang, Eric Juarez & Bobby G. Sumpter

npj Computational Materials **7**, Article number: 84 (2021) | Cite this article

The next big thing: the use of graph neural networks to discover particles

September 24, 2020 | Zack Savitsky

Share Tweet Email

Machine learning algorithms can beat the world's hardest video games in minutes and solve complex equations faster than the collective efforts of generations of physicists. But the conventional algorithms still struggle to pick out stop signs on a busy street.

Object identification continues to hamper the field of machine learning — especially when the pictures are multidimensional and complicated, like the ones particle detectors take of collisions in high-energy physics experiments. However, a new class of neural networks is helping these models boost their pattern recognition abilities, and the technology may soon be implemented in particle physics experiments to optimize data analysis.

nature View all journals Search Login

Explore content ⌄ About the journal ⌄ Publish with us ⌄

nature > articles > article

Article | Published: 09 June 2021

A graph placement methodology for fast chip design

Azalia Mirhoseini, Anna Goldie, Mustafa Yazgan, Joe Wenjie Jiang, Ebrahim Songhori, Shen Wang, Young-Joon Lee, Eric Johnson, Omkar Pathak, Azade Nazi, Jiwoo Pak, Andy Tong, Kavya Srinivasa, William Hang, Emre Tuncer, Quoc V. Le, James Laudon, Richard Ho, Roger Carpenter & Jeff Dean

Graph Neural Networks have a large impact on…

nature

Explore content ⌄ About the journal ⌄ Publish with us ⌄ Subscribe

nature > news > article

NEWS | 01 December 2021

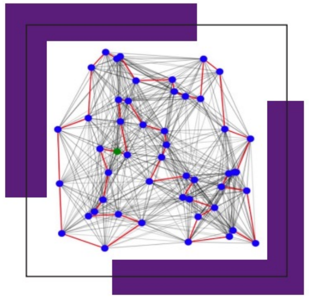
DeepMind's AI helps untangle the mathematics of knots

The machine-learning techniques could sets.

ipam institute for pure & applied mathematics

Deep Learning and Combinatorial Optimization

February 22 - 25, 2021



Patterns

CellPress OPEN ACCESS

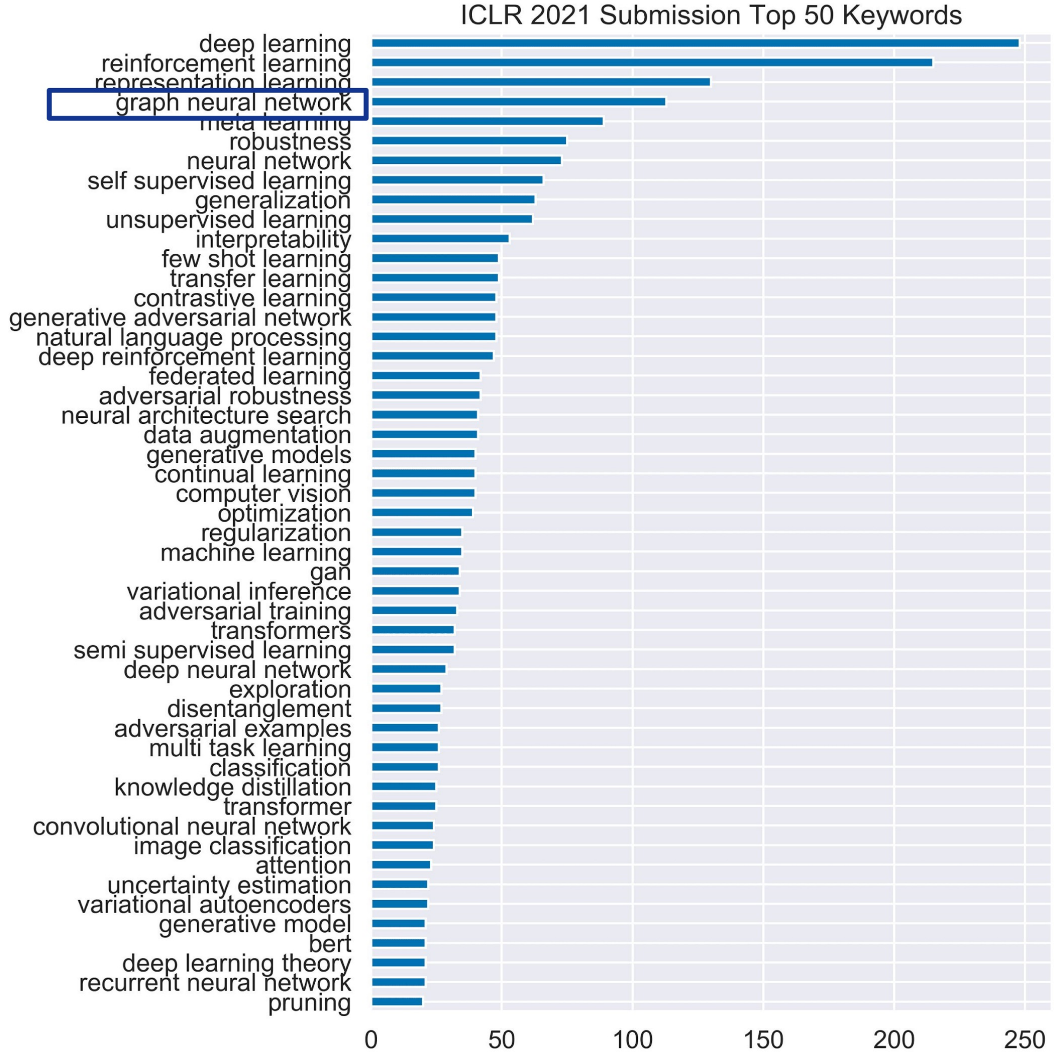
Opinion

Neural algorithmic reasoning

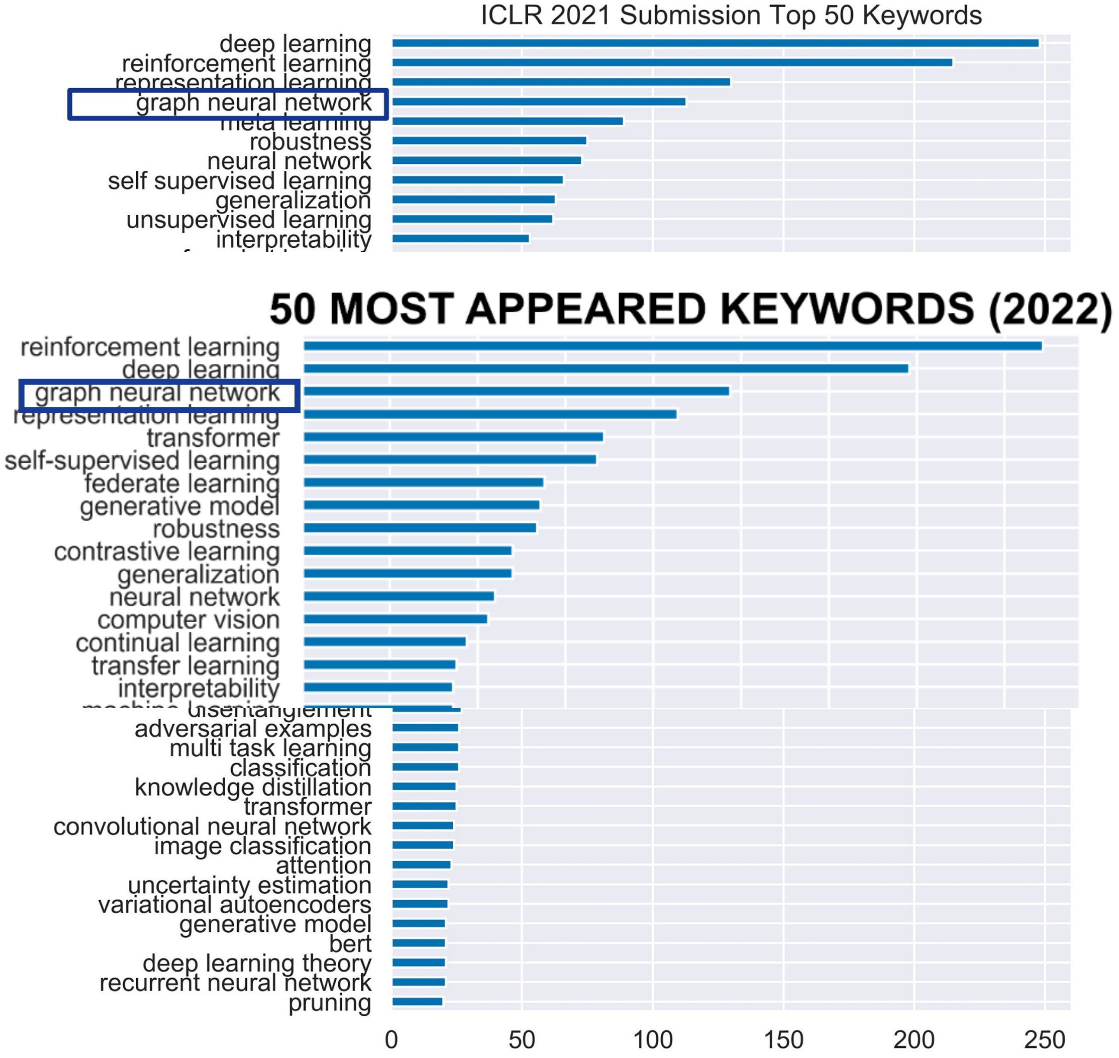
Petar Veličković[1,*] and Charles Blundell[1]
[1]DeepMind, London, Greater London, UK
*Correspondence: petarv@google.com
https://doi.org/10.1016/j.patter.2021.100273

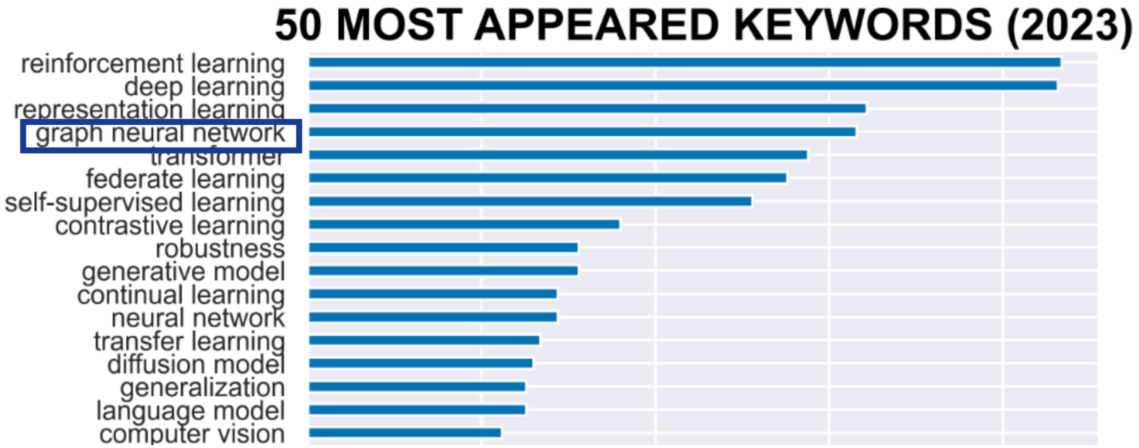
We present neural algorithmic reasoning—the art of building neural networks that are able to execute algorithmic computation—and provide our opinion on its transformative potential for running classical algorithms on inputs previously considered inaccessible to them.

Popular research topic

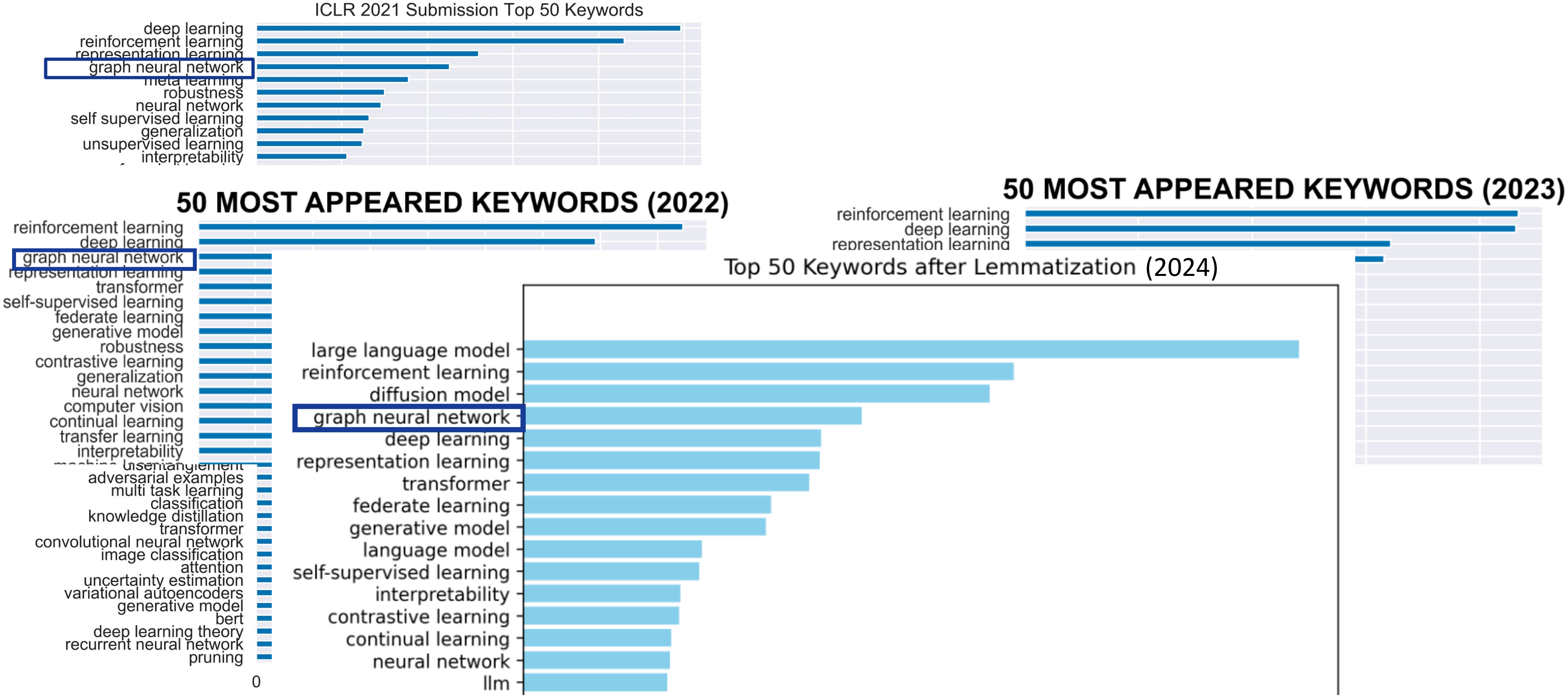


Popular research topic



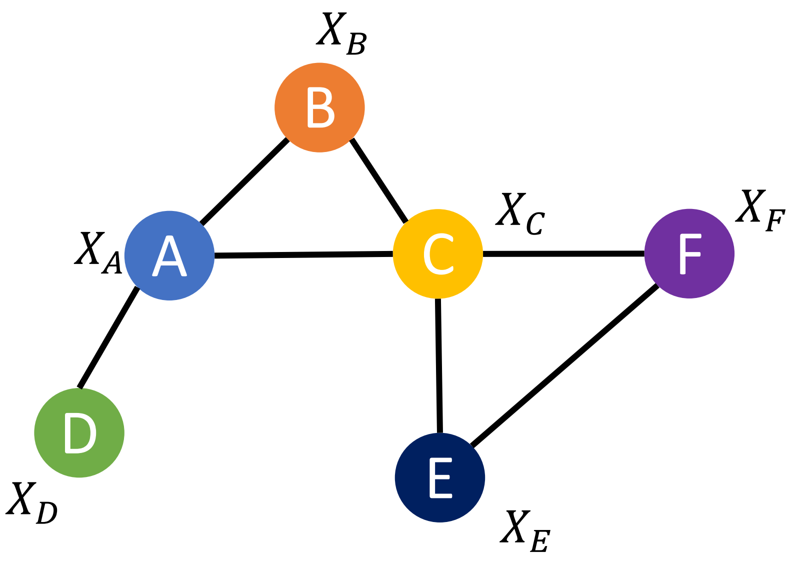


Popular research topic



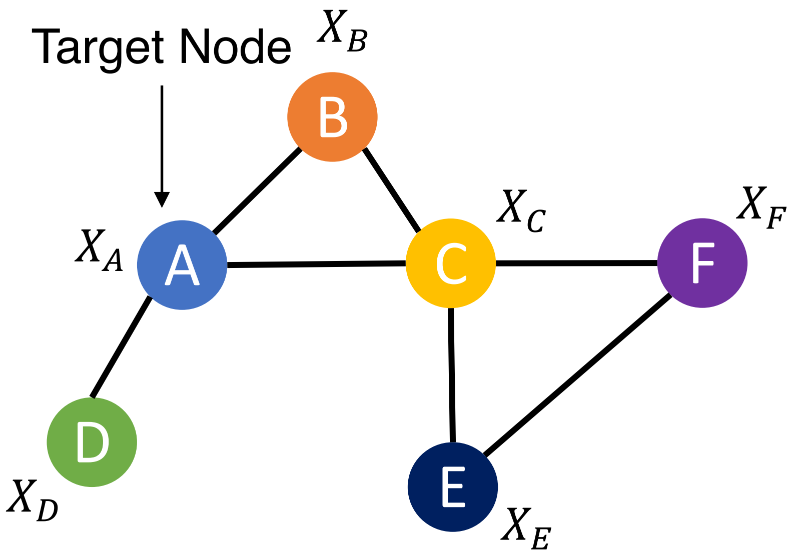
What is Graph Neural Network?

Problem definition



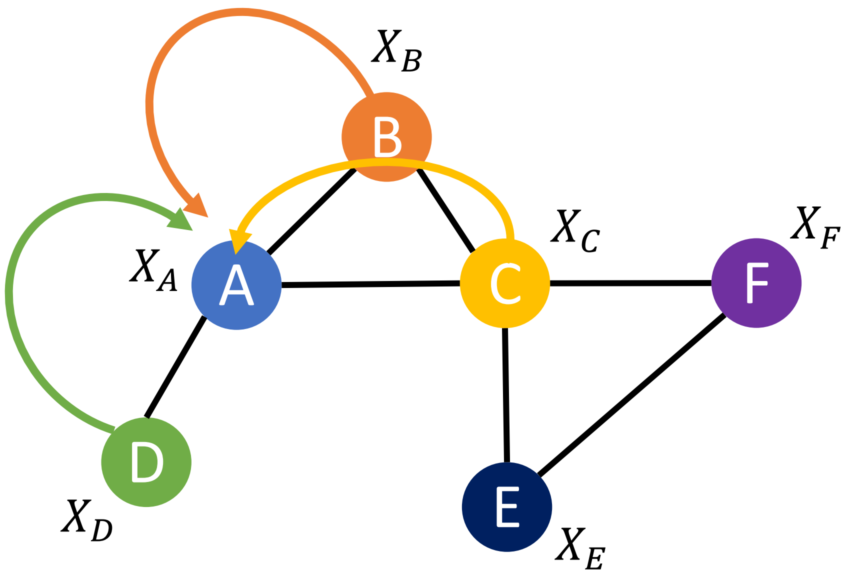
- **Given**
  - A graph
  - Node attributes
  - (part of nodes are labeled)
- **Find**
  - Node embeddings
- **Predict**
  - Labels for the remaining nodes

Graph Neural Networks



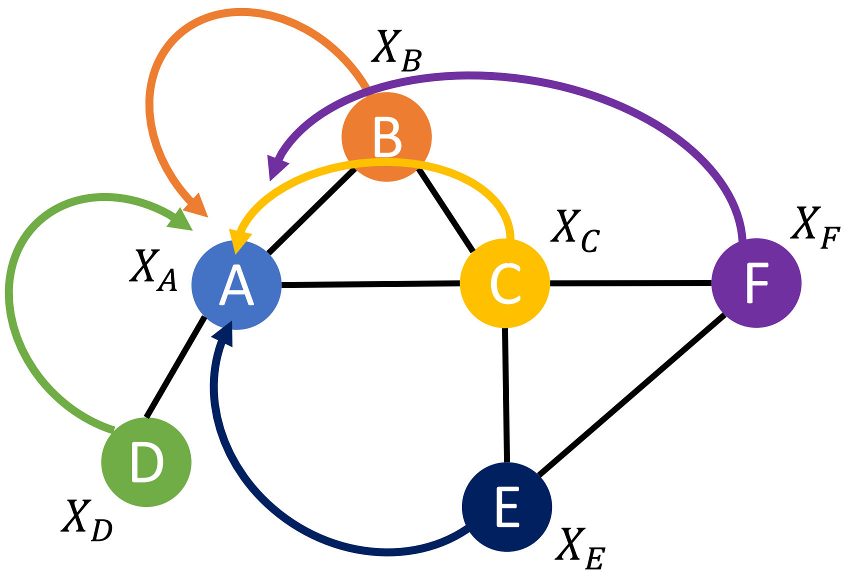
"Homophily: connected nodes are related/informative/similar"

Graph Neural Networks



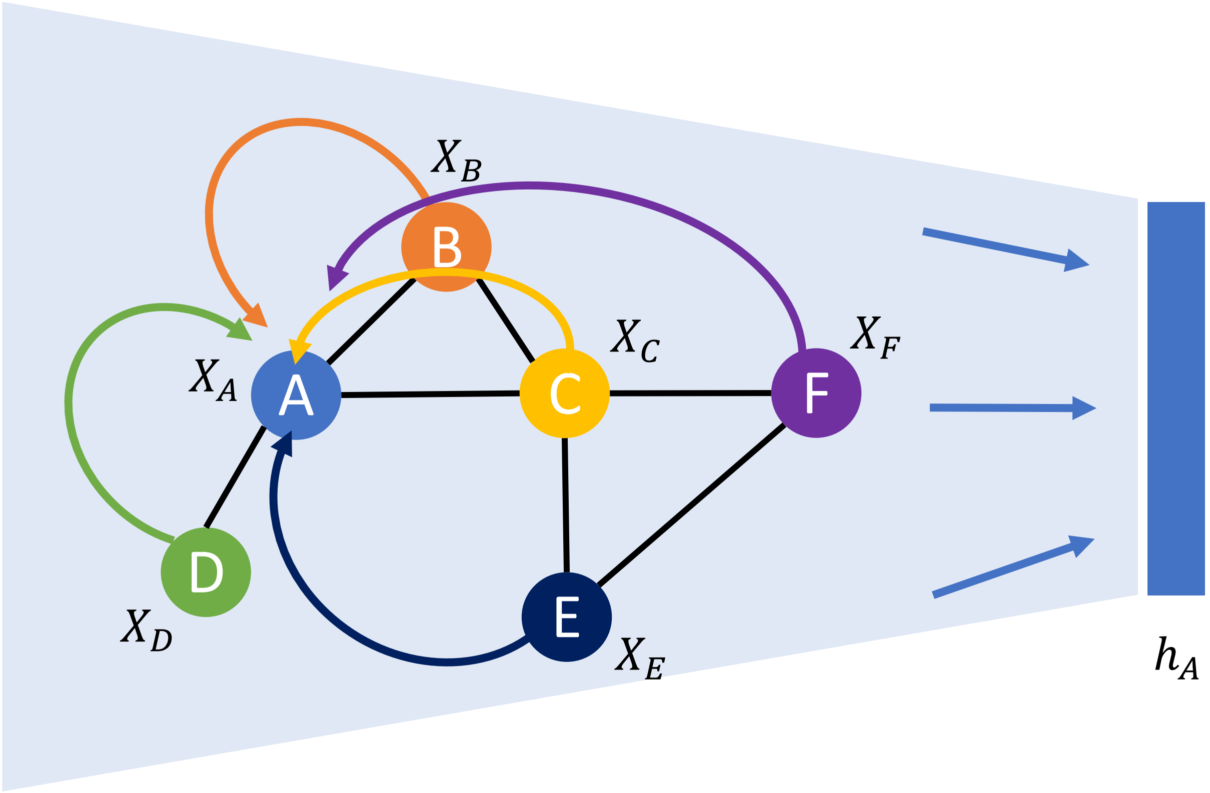
“Homophily: connected nodes are related/informative/similar”

Graph Neural Networks

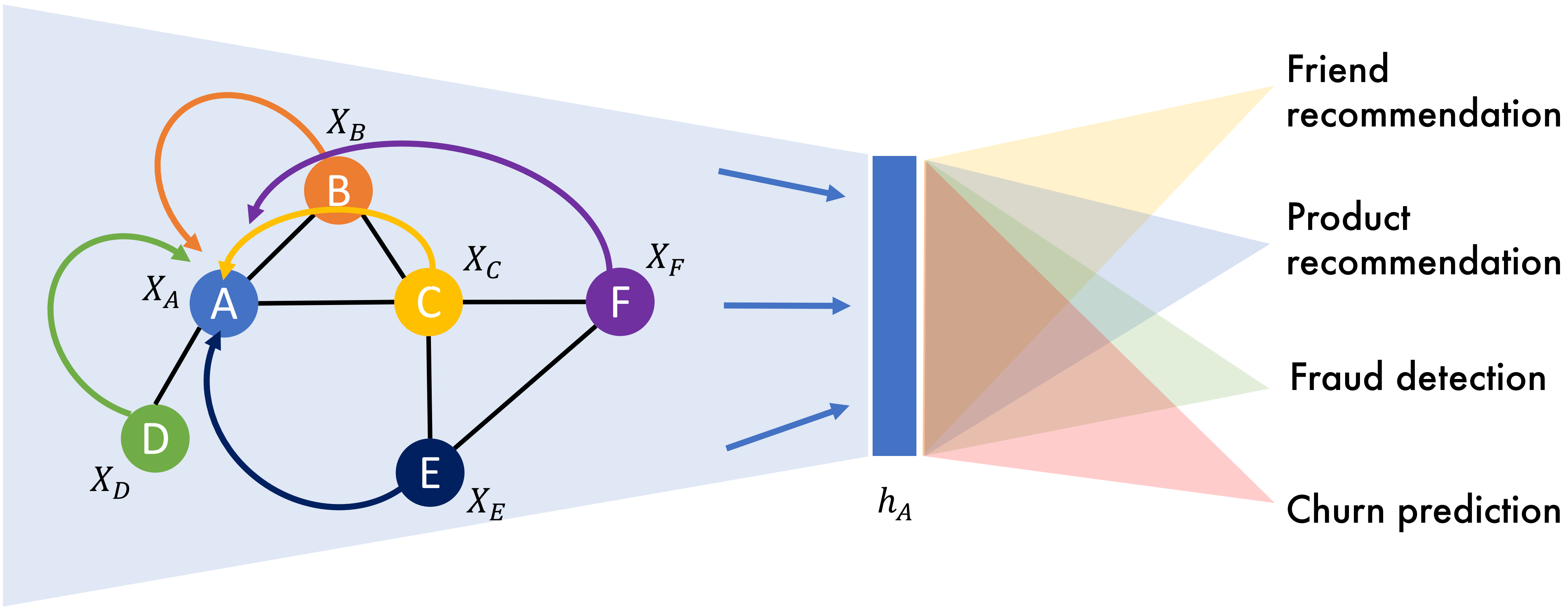


"Homophily: connected nodes are related/informative/similar"

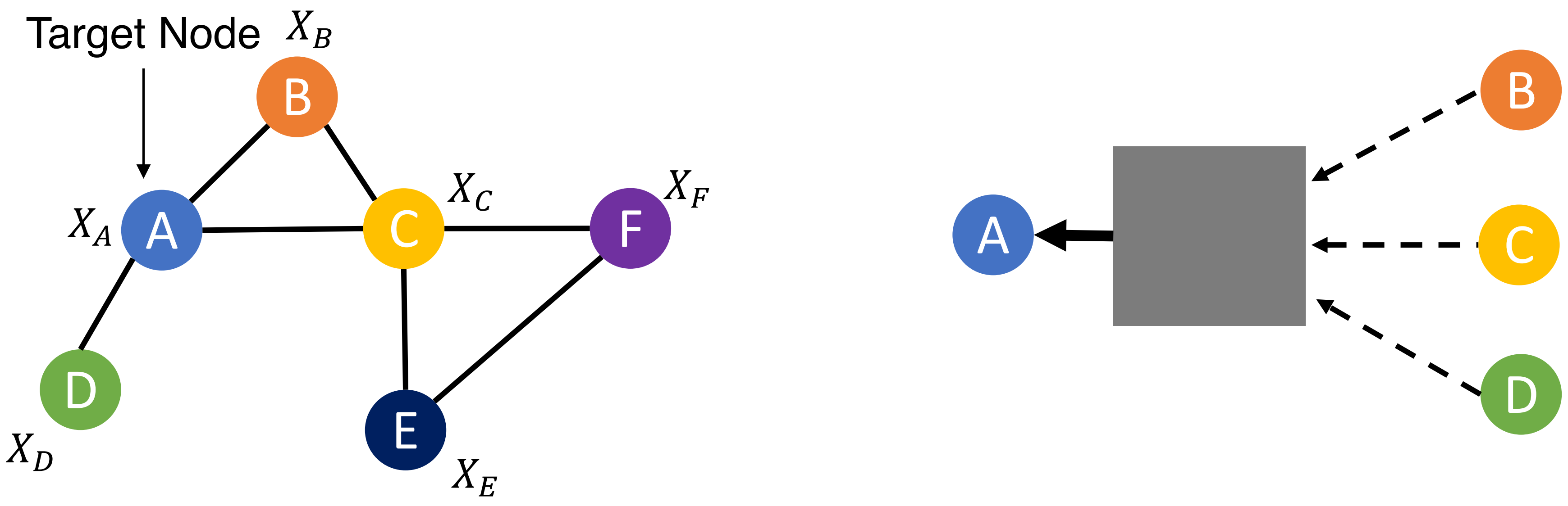
Graph Neural Networks



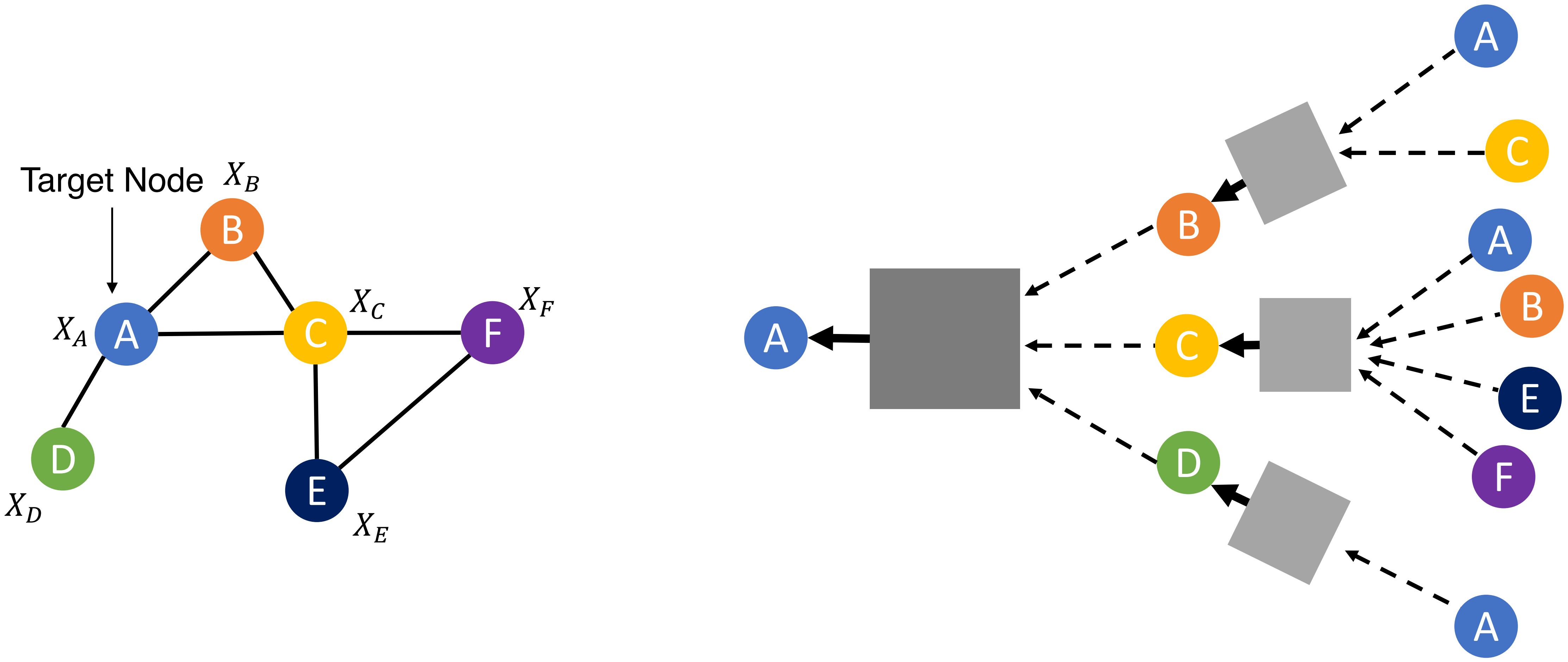
Graph Neural Networks



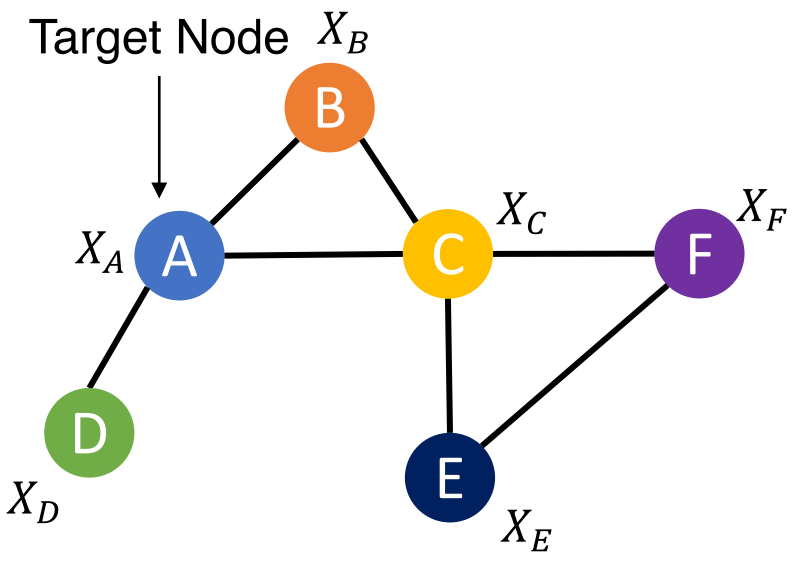
Graph Neural Networks

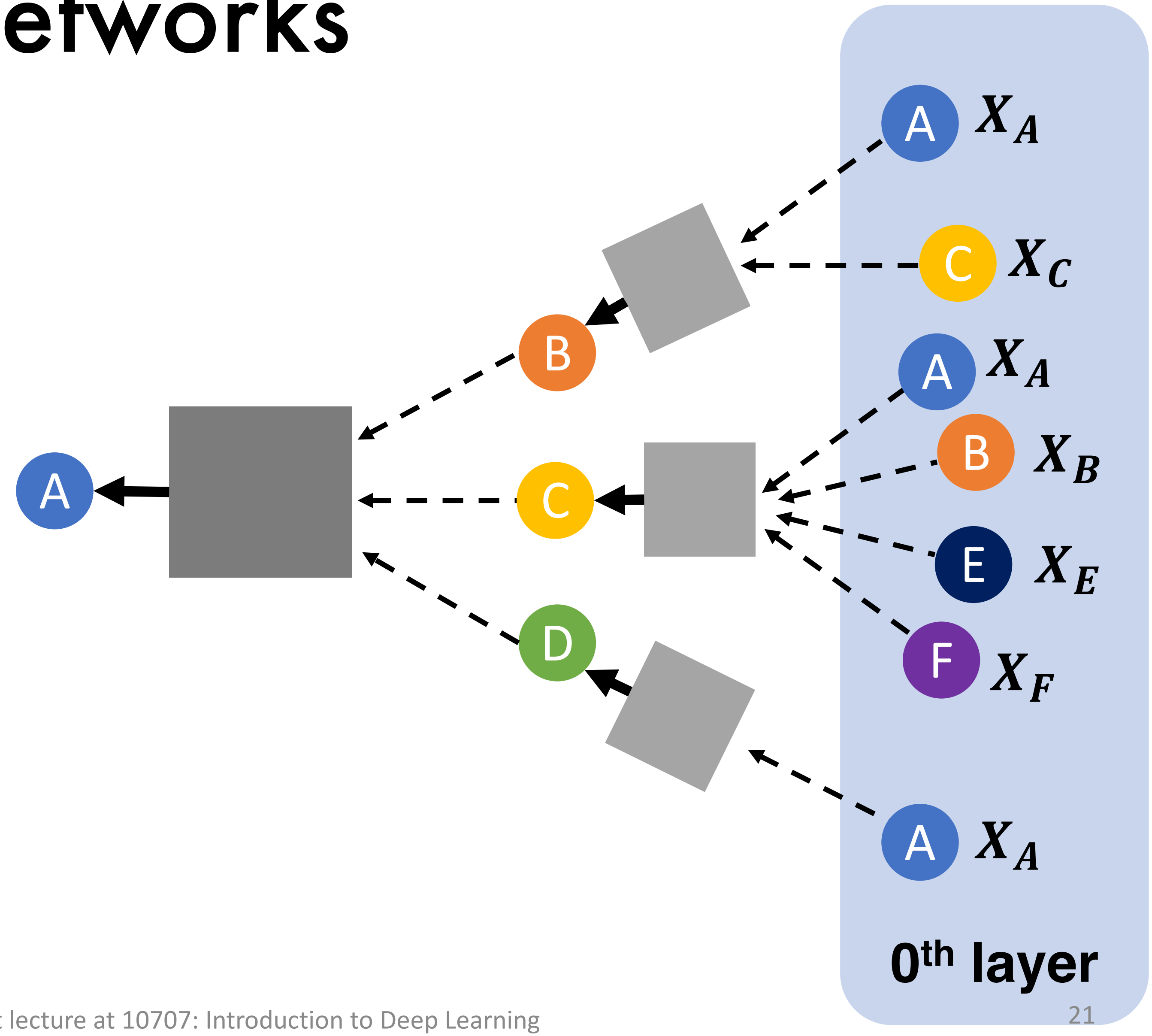


Graph Neural Networks

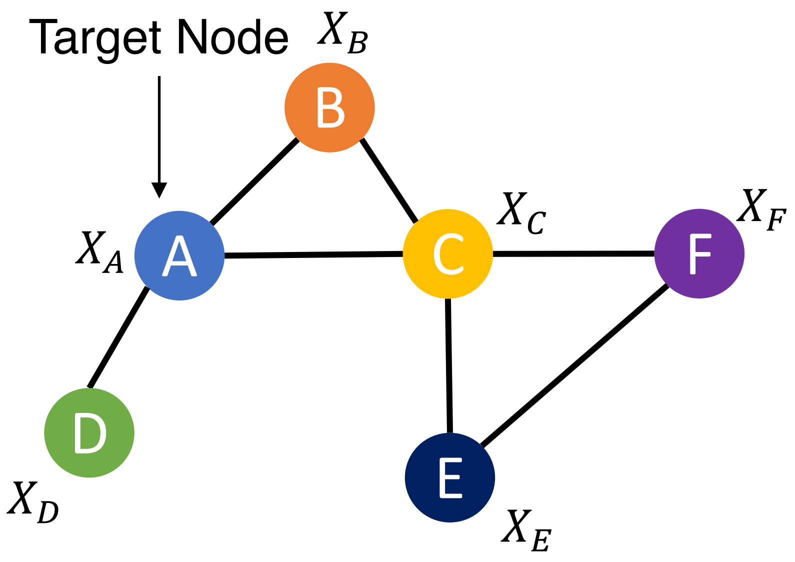


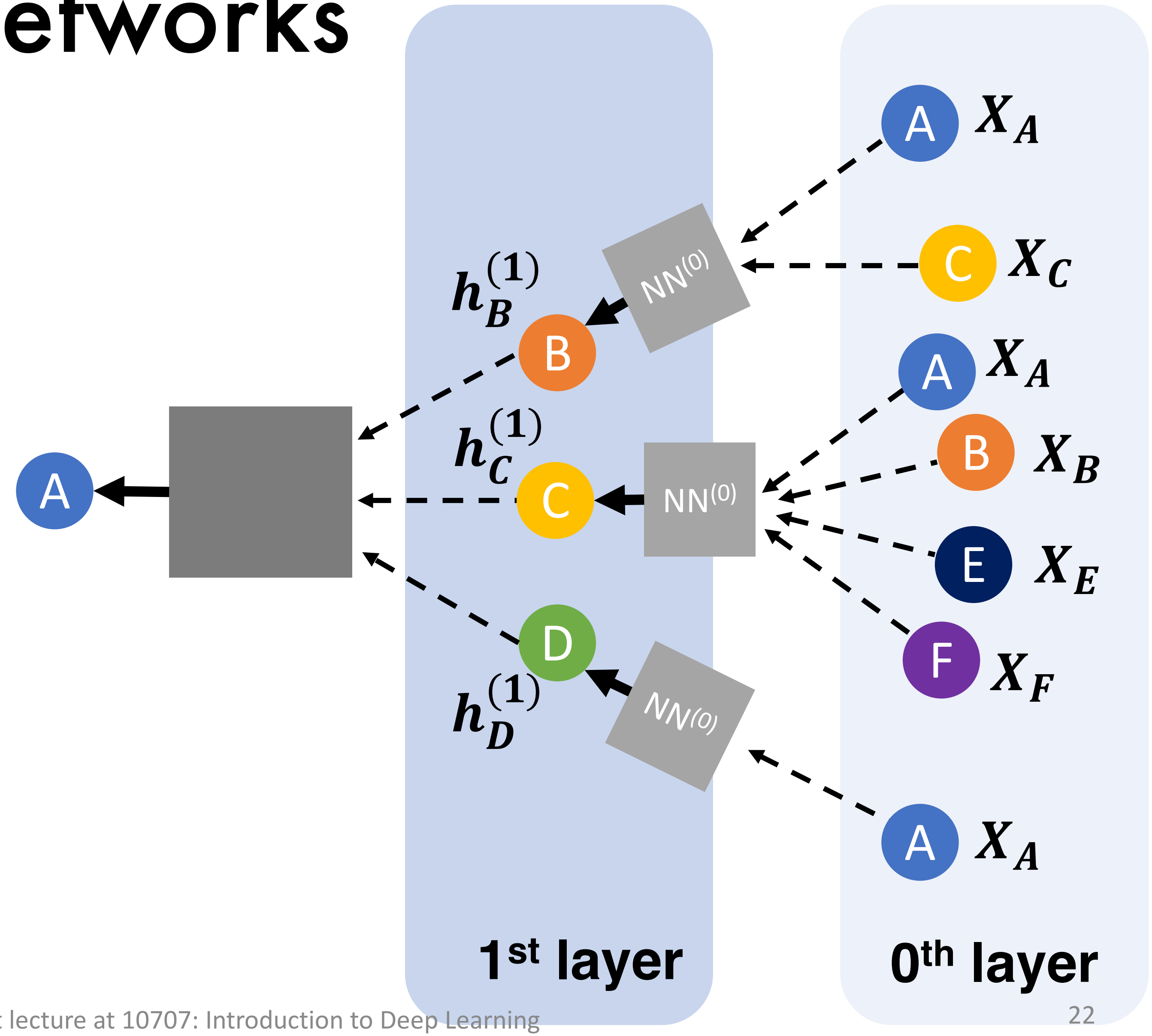
Graph Neural Networks



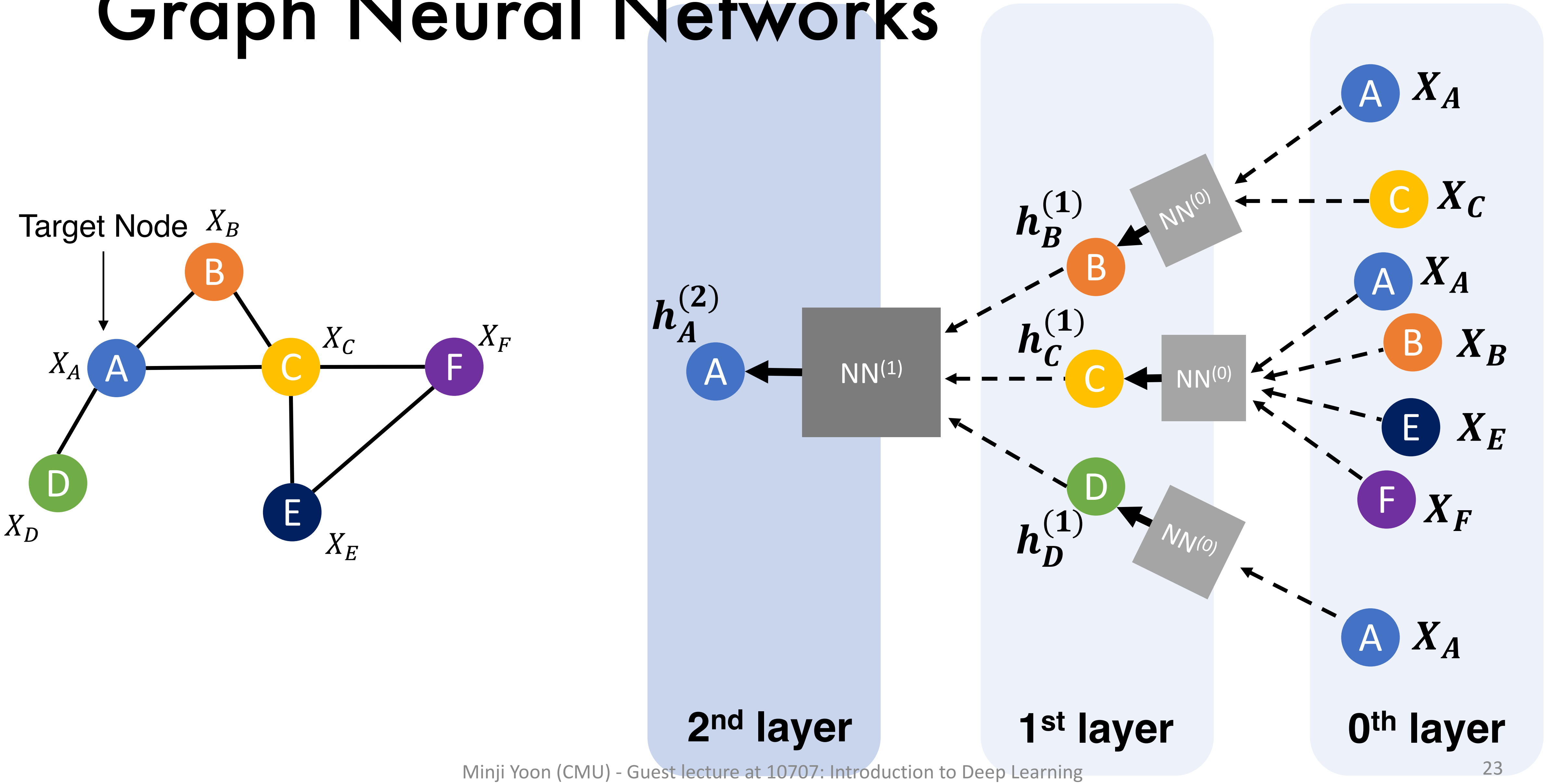


Graph Neural Networks





Graph Neural Networks

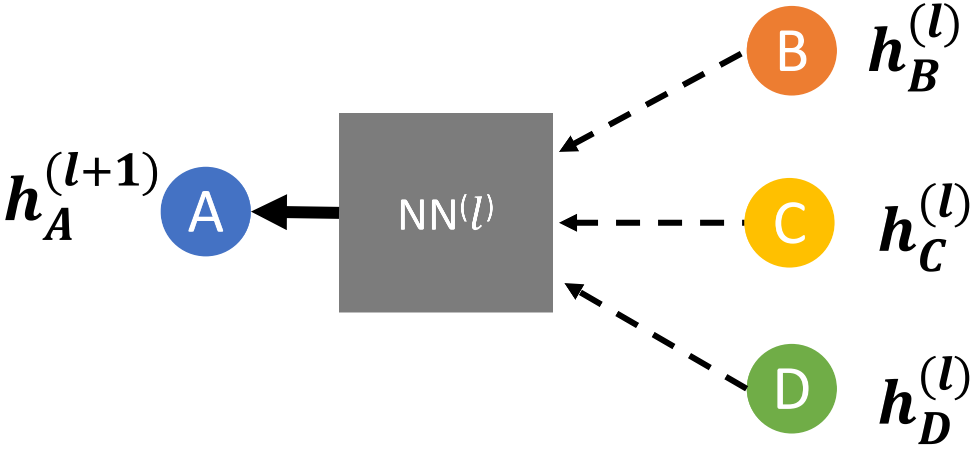


Graph Neural Networks

1. **Aggregate messages from neighbors**

$h_v^{(l)}$: node embedding of v at l-th layer
$\mathcal{N}(v)$: neighboring nodes of v
$\boldsymbol{f}^{(l)}$: aggregation function at l-th layer
$m_v^{(l)}$: message vector of v at l-th layer

$$m_A^{(l)} = \boldsymbol{f}^{(l)}\left(h_A^{(l)}, \left\{h_u^{(l)} : u \in \mathcal{N}(A)\right\}\right)$$
$$= \boldsymbol{f}^{(l)}\left(h_A^{(l)}, h_B^{(l)} h_C^{(l)} h_D^{(l)}\right)$$



Neighbors of node A
$\mathcal{N}(A) = \{B, C, D\}$

Graph Neural Networks

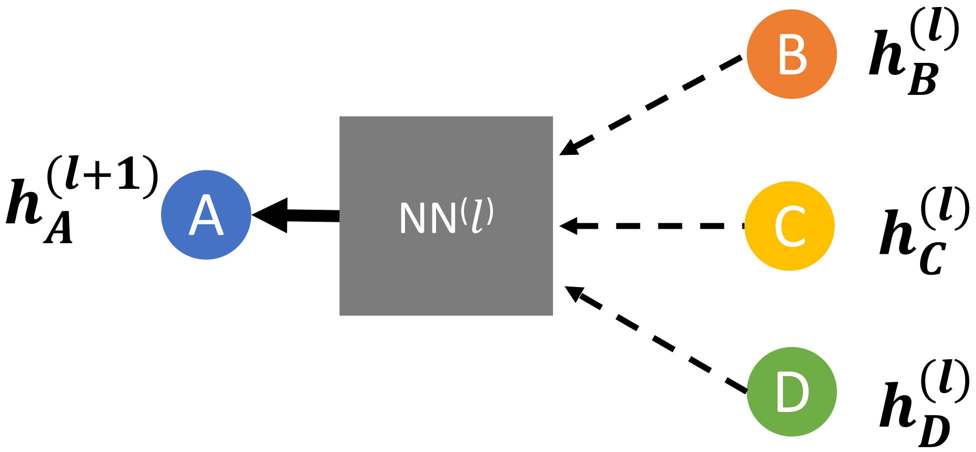
1. Aggregate messages from neighbors

$$m_A^{(l)} = \boldsymbol{f}^{(l)}\left(h_A^{(l)}, \left\{h_u^{(l)} : u \in \mathcal{N}(A)\right\}\right)$$
$$= \boldsymbol{f}^{(l)}\left(h_A^{(l)}, h_B^{(l)} h_C^{(l)} h_D^{(l)}\right)$$

2. Transform messages

$\boldsymbol{g}^{(l)}$: transformation function at l-th layer

$$h_A^{(l+1)} = \boldsymbol{g}^{(l)}(m_A^{(l)})$$



Neighbors of node A
$\mathcal{N}(A) = \{B, C, D\}$

Graph Neural Networks

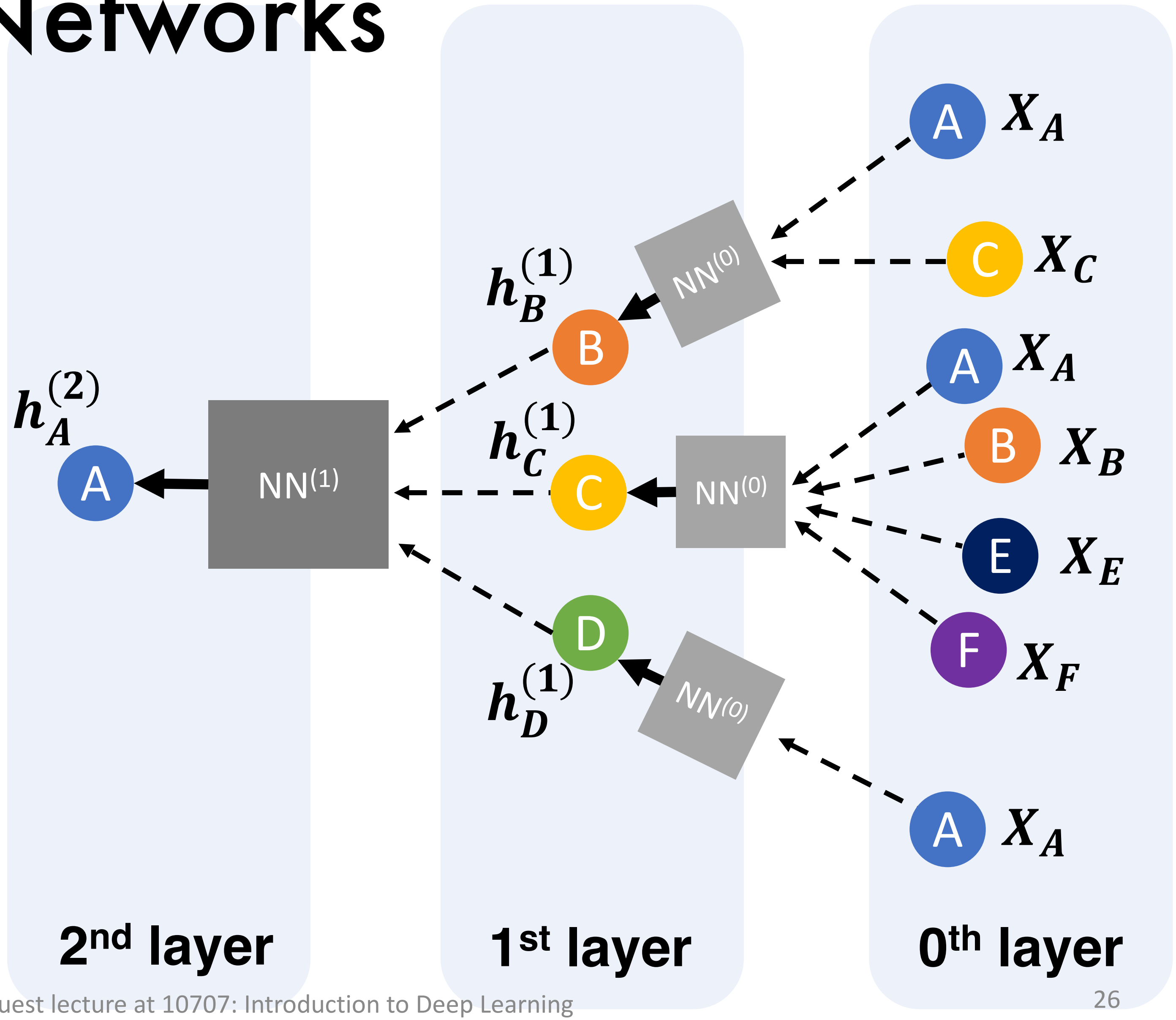
In each layer l ,
for each target node v :

1. Aggregate messages

$m_v^{(l)} = \boldsymbol{f}^{(l)}\left(h_v^{(l)}, \left\{h_u^{(l)} : u \in \mathcal{N}(v)\right\}\right)$

2. Transform messages

$h_v^{(l+1)} = \boldsymbol{g}^{(l)}(m_v^{(l)})$



Graph Neural Networks

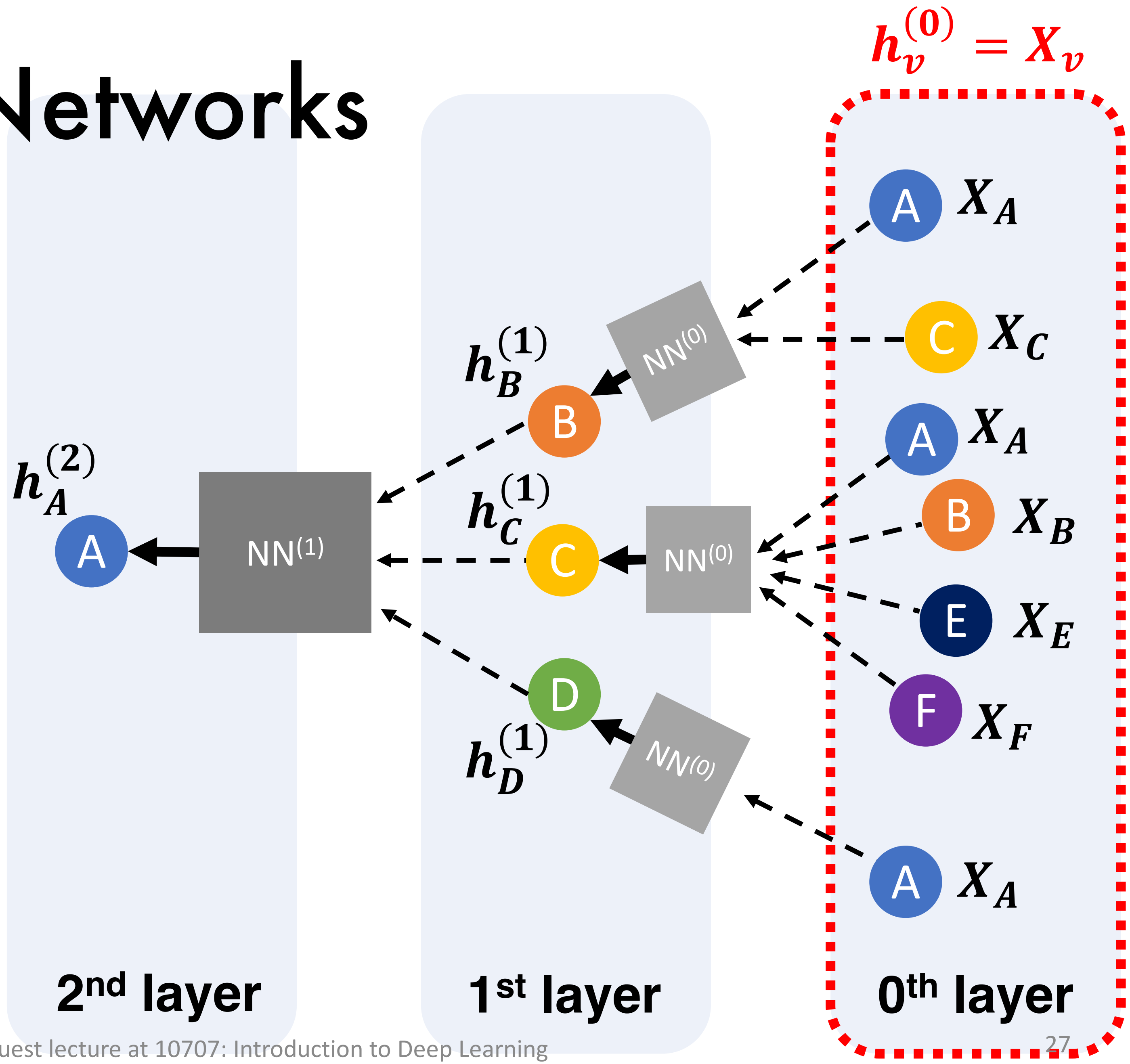
In each layer l ,
for each target node v :

1. Aggregate messages

$$m_v^{(l)} = \boldsymbol{f}^{(l)}\left(h_v^{(l)}, \left\{h_u^{(l)} : u \in \mathcal{N}(v)\right\}\right)$$

2. Transform messages

$$h_v^{(l+1)} = \boldsymbol{g}^{(l)}(m_v^{(l)})$$



Graph Neural Networks

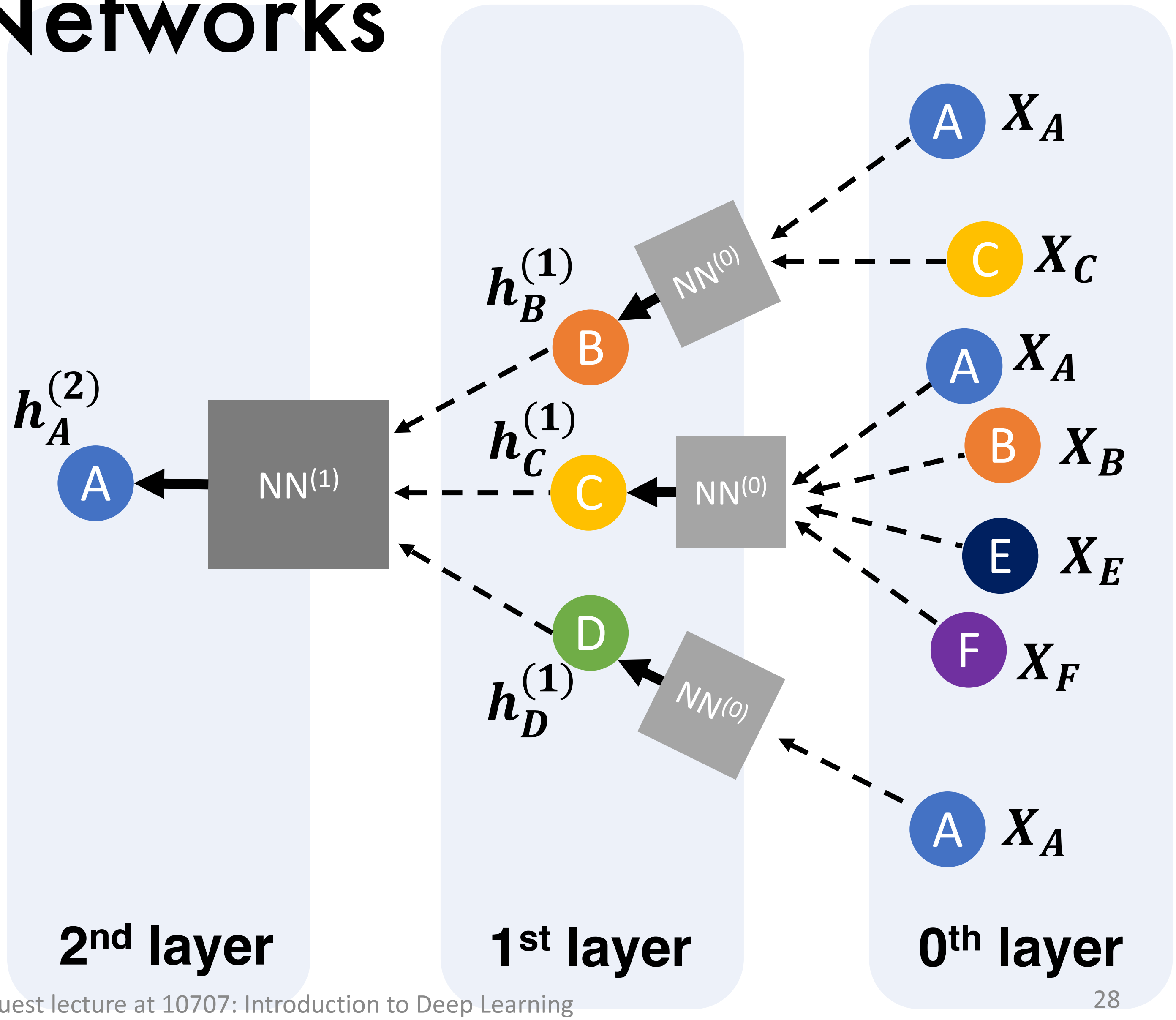
In each layer l ,
for each target node v :

1. Aggregate messages

$$m_v^{(l)} = \boxed{\boldsymbol{f}^{(l)}}\left(h_v^{(l)}, \left\{h_u^{(l)} : u \in \mathcal{N}(v)\right\}\right)$$

2. Transform messages

$$h_v^{(l+1)} = \boxed{\boldsymbol{g}^{(l)}}(m_v^{(l)})$$



Graph Neural Networks

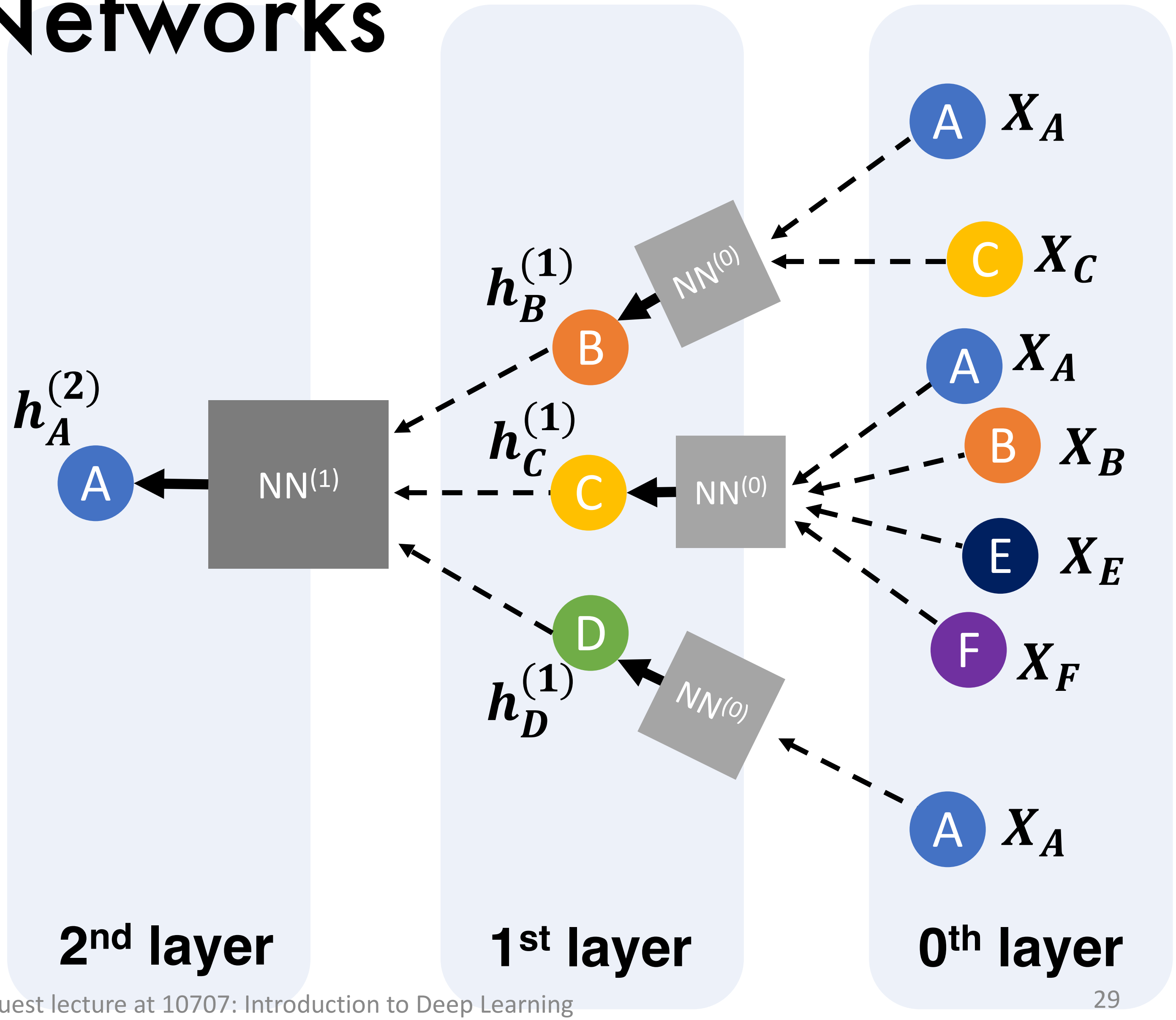
Graph Convolutional Networks[1]

1. Aggregate messages

$$m_v^{(l)} = \frac{1}{|\mathcal{N}(v) + 1|} \sum_{u \in \mathcal{N}(v) \cup \{v\}} h_u^{(l)}$$

2. Transform messages

$$h_v^{(l+1)} = \sigma(\boldsymbol{W}^{(l)} \circ m_v^{(l)})$$



[1] Kipf, Thomas N., et al. "Semi-supervised classification with graph convolutional networks."

Graph Neural Networks

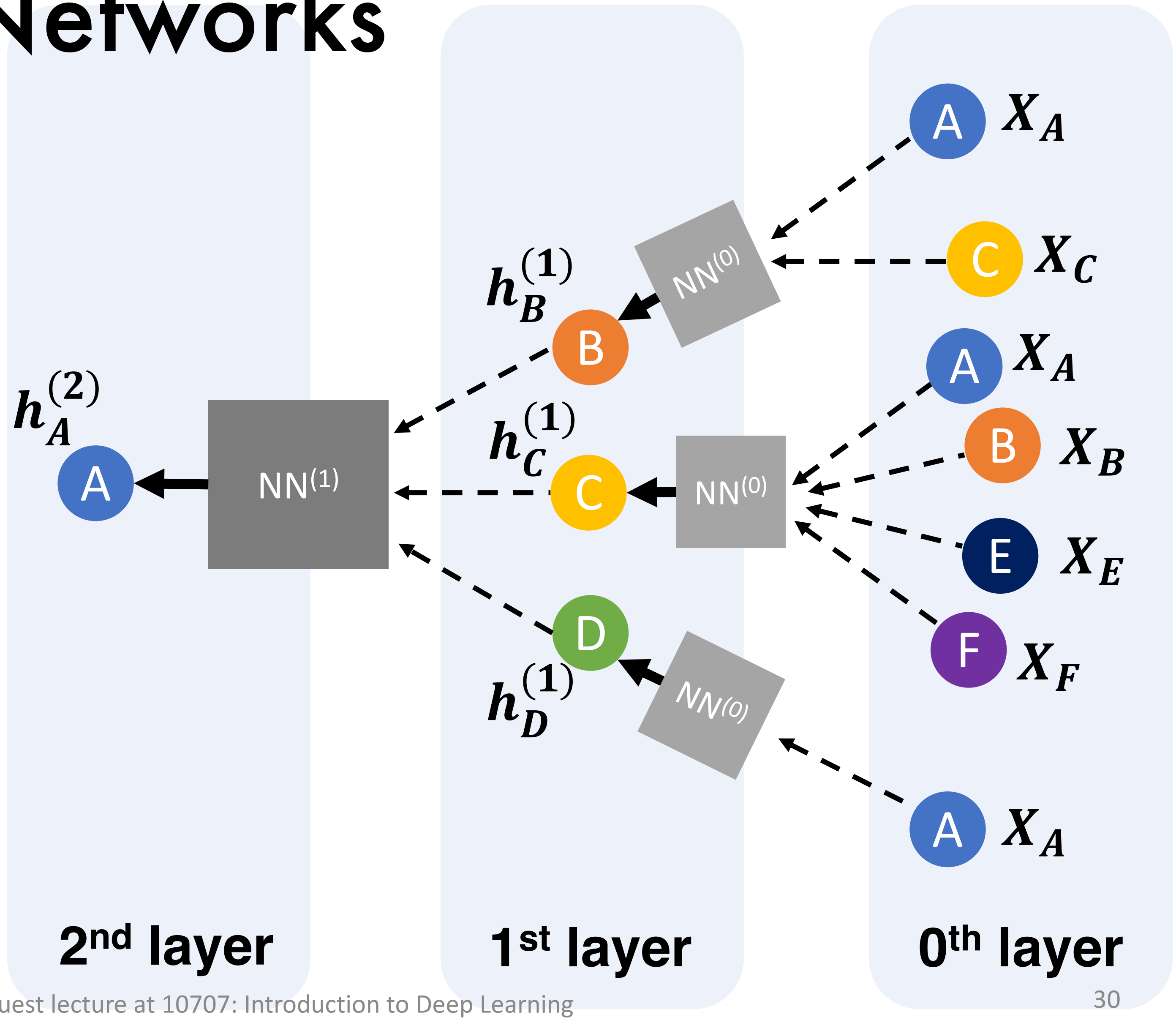
Graph Isomorphism Networks[2]

1. Aggregate messages

$$m_v^{(l)} = \sum_{u \in \mathcal{N}(v) \cup \{v\}} h_u^{(l)}$$

2. Transform messages

$$h_v^{(l+1)} = \sigma(\boldsymbol{W}^{(l)} \circ m_v^{(l)})$$



[2] Xu, Keyulu, et al. "How powerful are graph neural networks?."

Graph Neural Networks

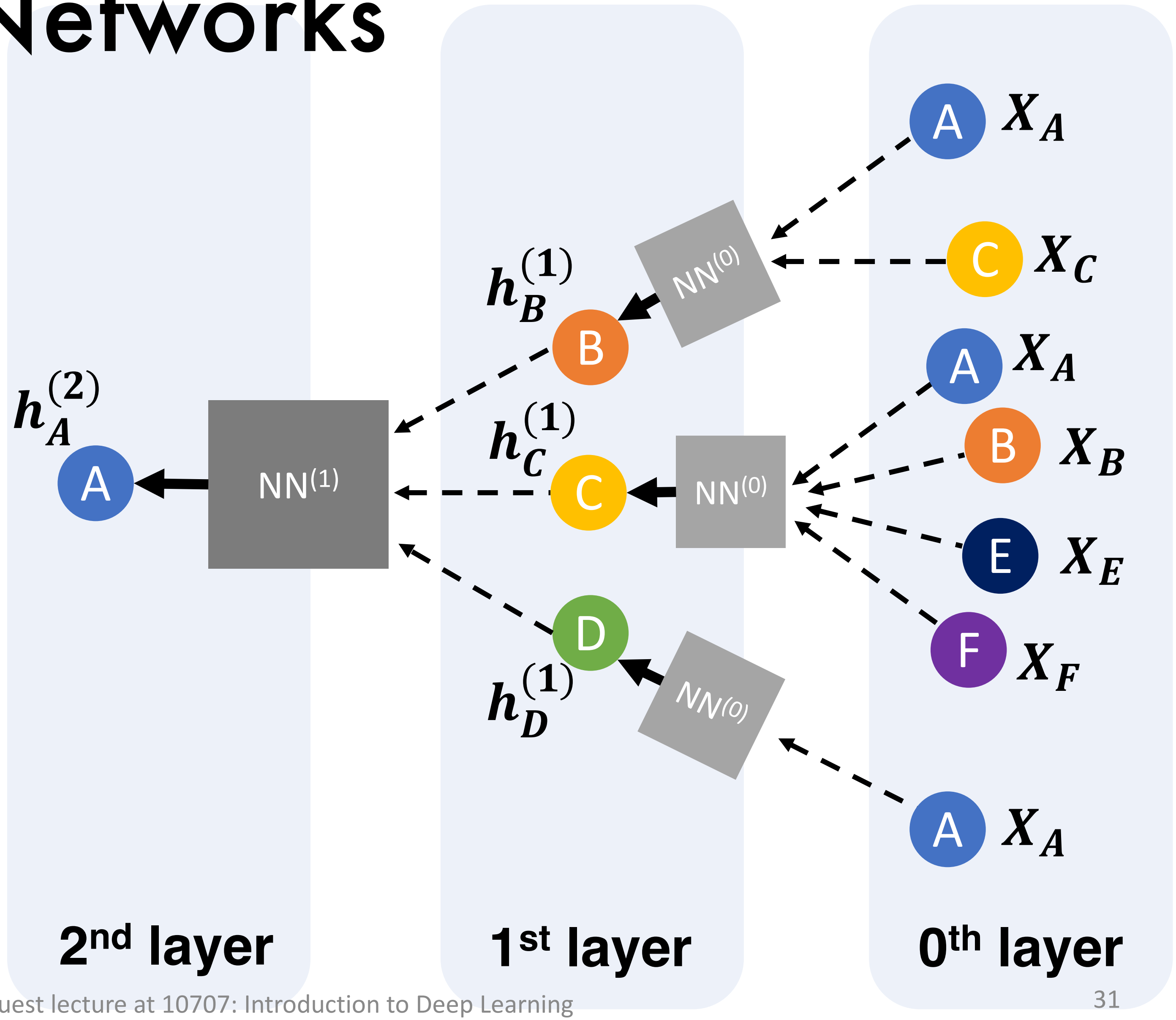
Simplified GCN[3]

1. Aggregate messages

$$m_v^{(l)} = \frac{1}{|\mathcal{N}(v) + 1|} \sum_{u \in \mathcal{N}(v) \cup \{v\}} h_u^{(l)}$$

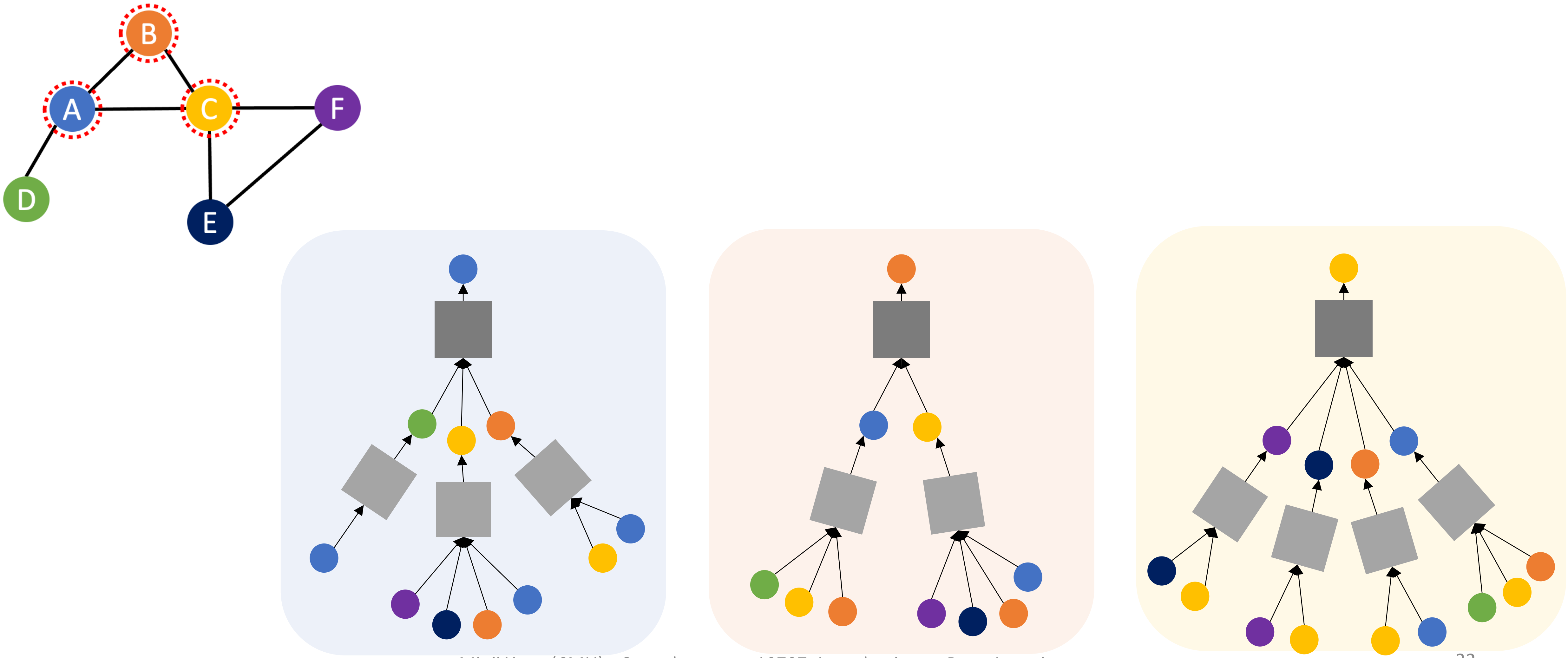
2. Transform messages

$$h_v^{(l+1)} = \boldsymbol{W}^{(l)} \circ m_v^{(l)}$$

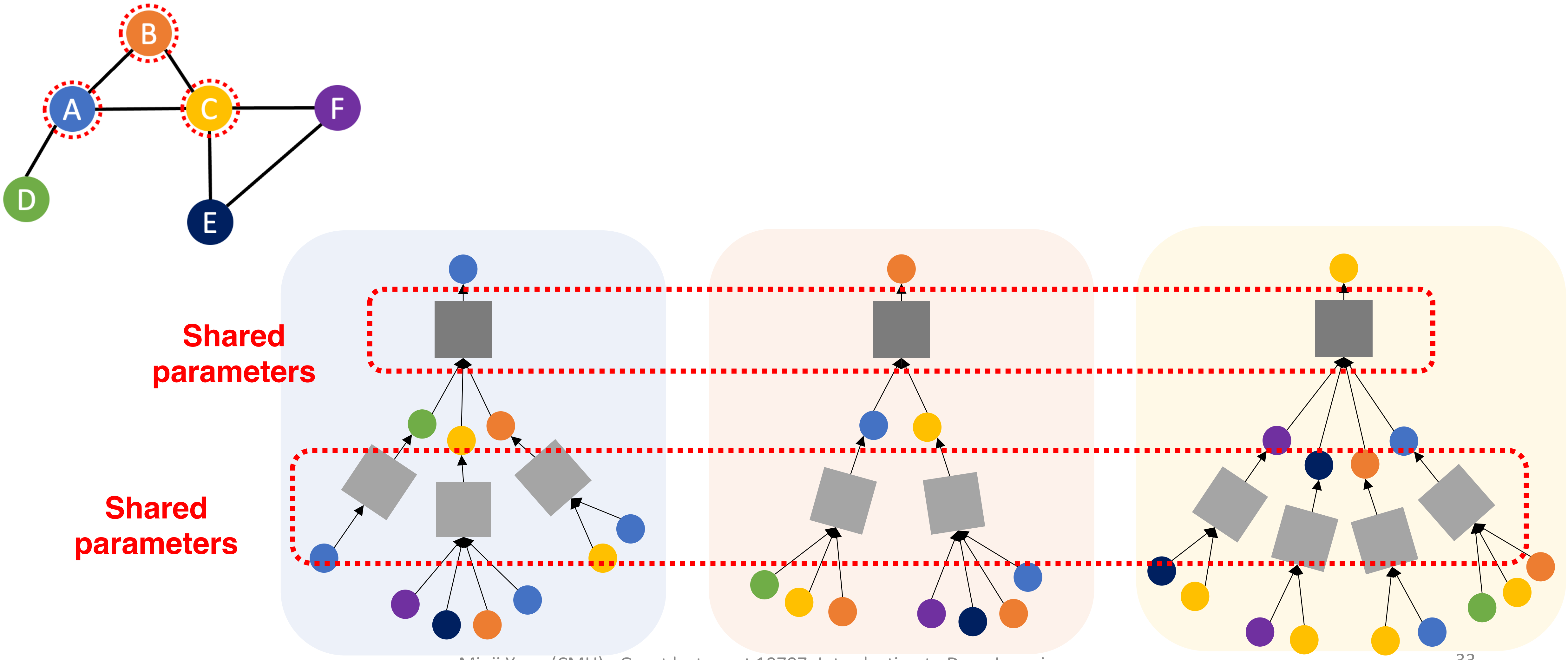


[3] Wu, Felix, et al. "Simplifying graph convolutional networks."

Computation graphs

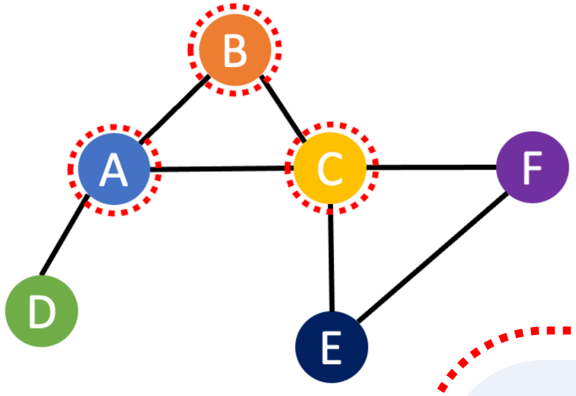


Computation graphs

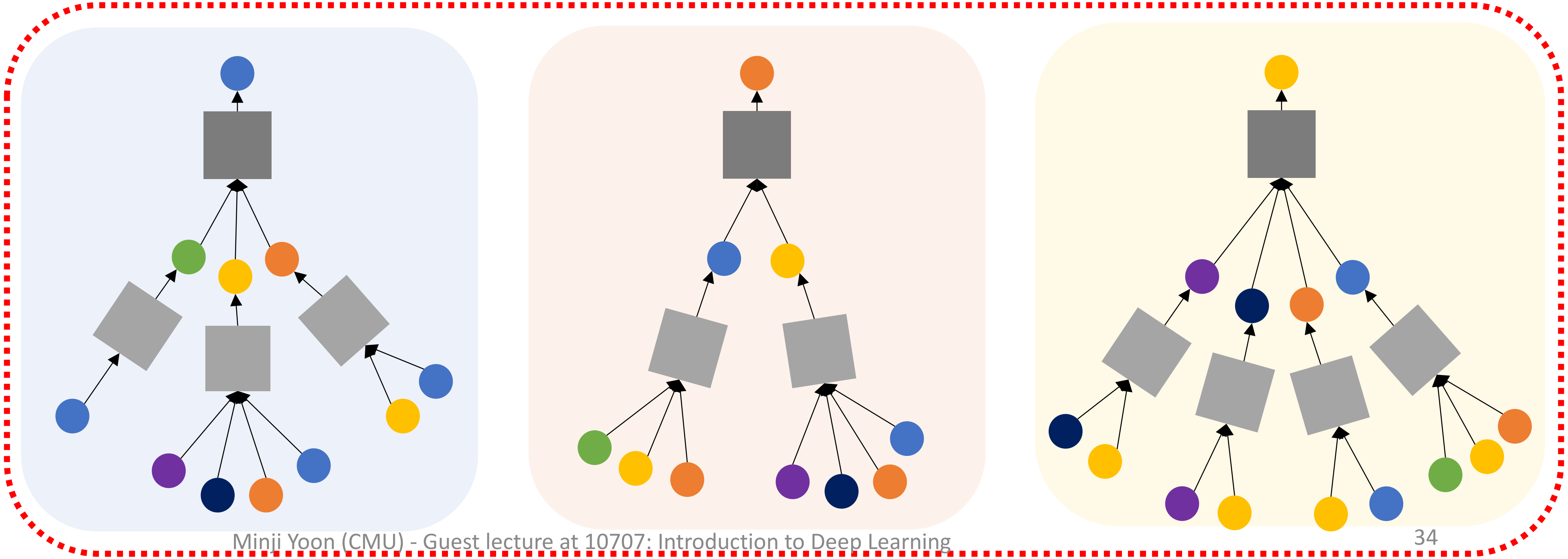


Batch execution

$$h_v^{(l)} = \sigma(\boldsymbol{W}^{(l)} \circ (\frac{1}{|\mathcal{N}(v)+1|} \sum_{u \in \mathcal{N}(v) \cup \{v\}} h_u^{(l-1)}))$$



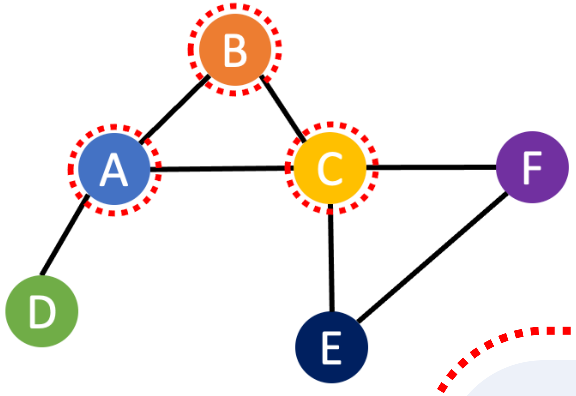
Batch size = 3



Batch execution

$$h_v^{(l)} = \sigma(\boldsymbol{W}^{(l)} \circ (\frac{1}{|\mathcal{N}(v)+1|}\sum_{u \in \mathcal{N}(v)\cup\{v\}} h_u^{(l-1)}))$$

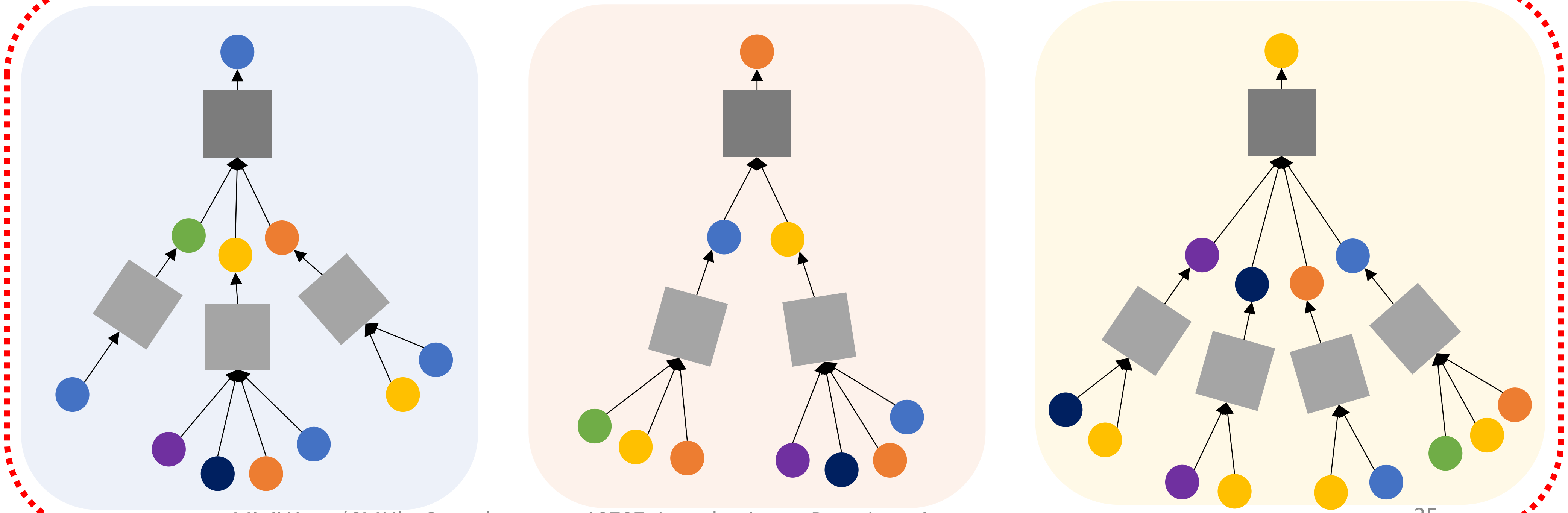
$$\boldsymbol{H}^{(l)} = \sigma(\widetilde{(\boldsymbol{A}+\boldsymbol{I})}\boldsymbol{H}^{(l-1)}\boldsymbol{W}^{(l)})$$



Node embedding matrix

(row-normalized) **Adjacency matrix**

Batch size = 3

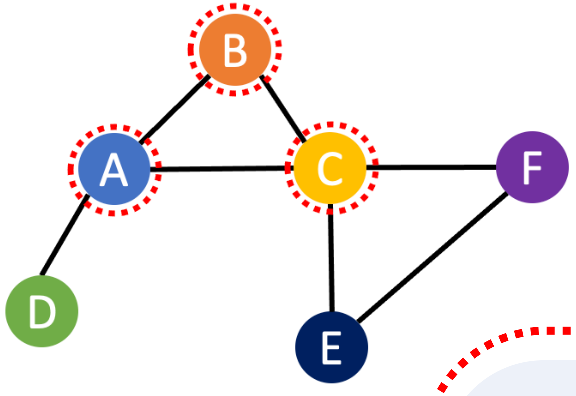


Batch execution

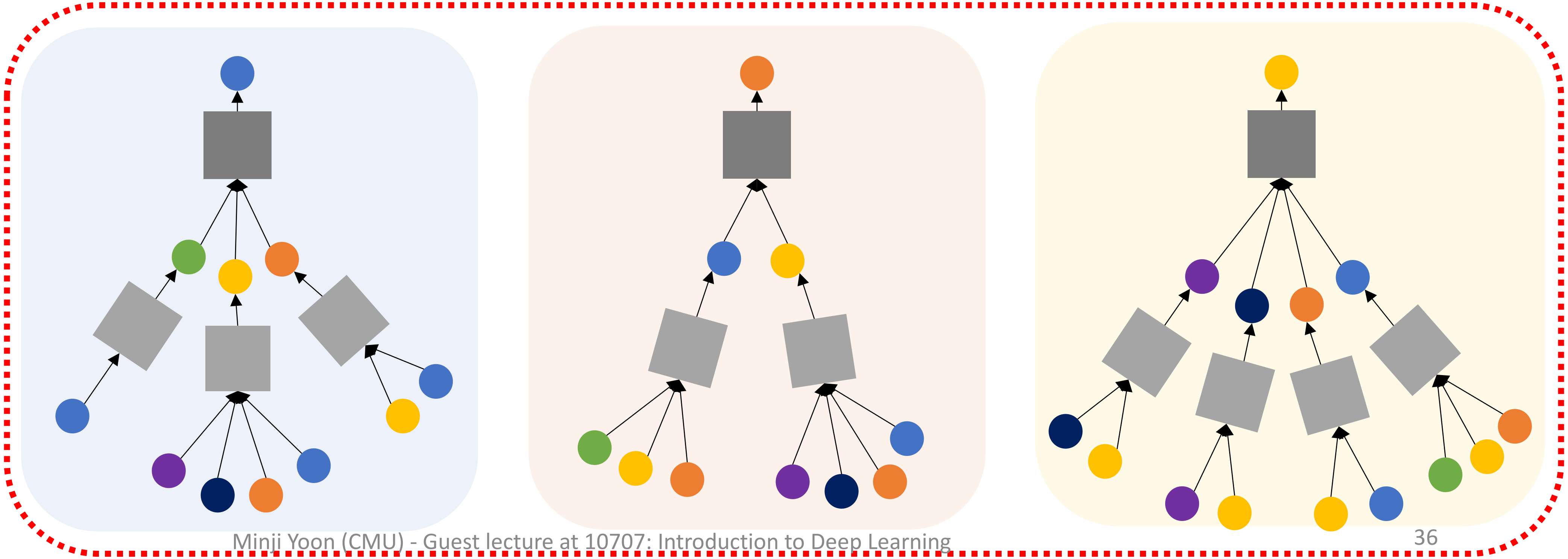
$$h_v^{(l)} = \sigma(\boldsymbol{W}^{(l)} \circ (\frac{1}{|\mathcal{N}(v)+1|}\sum_{u\in\mathcal{N}(v)\cup\{v\}} h_u^{(l-1)}))$$

$$\boldsymbol{H}^{(l)} = \sigma(\boxed{\widetilde{(\boldsymbol{A}+\boldsymbol{I})}}\boldsymbol{H}^{(l-1)}\boxed{\boldsymbol{W}^{(l)}})$$

Fixed **Trainable**

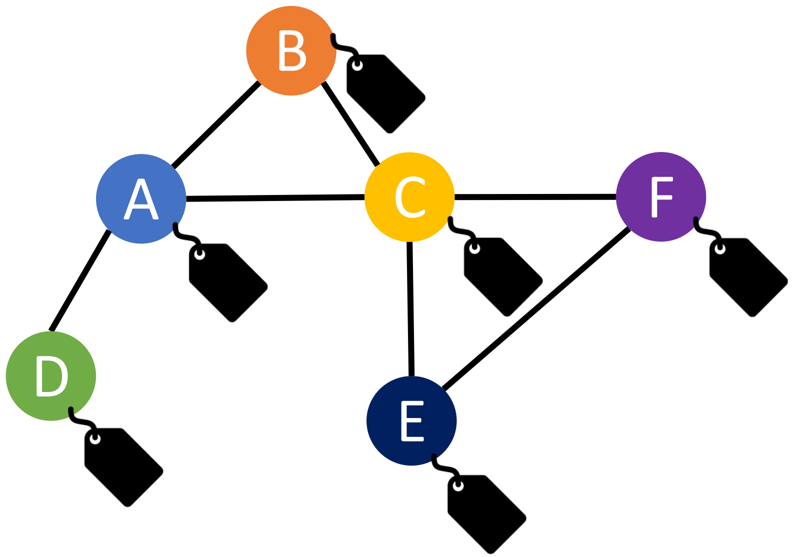


Batch size = 3



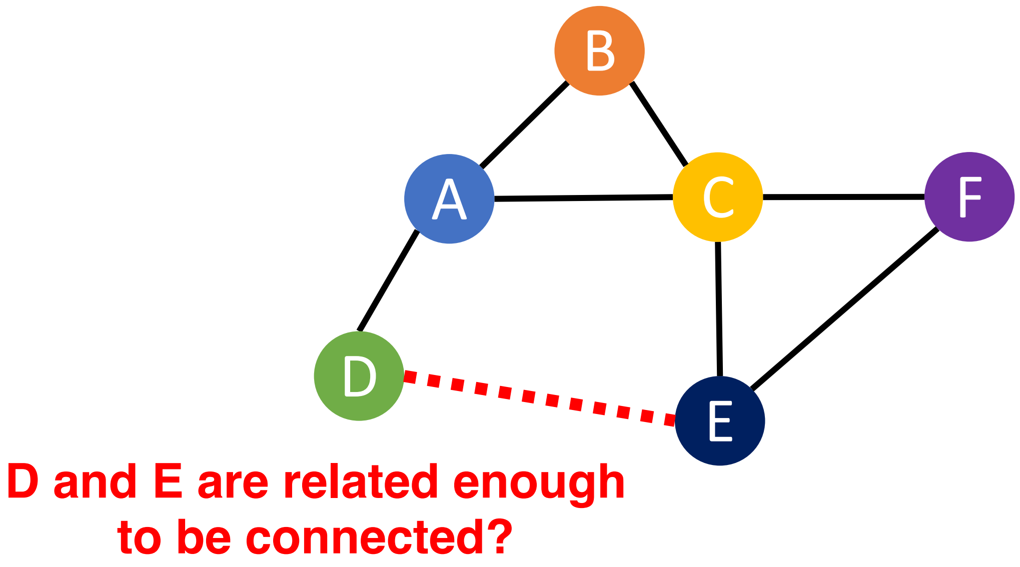
Downstream tasks

- Node-level prediction



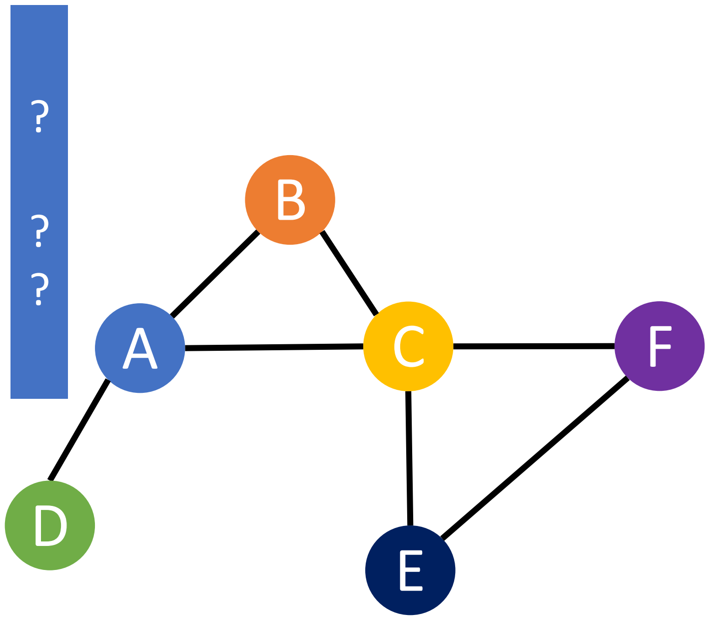
Downstream tasks

- Node-level prediction
- Edge-level prediction



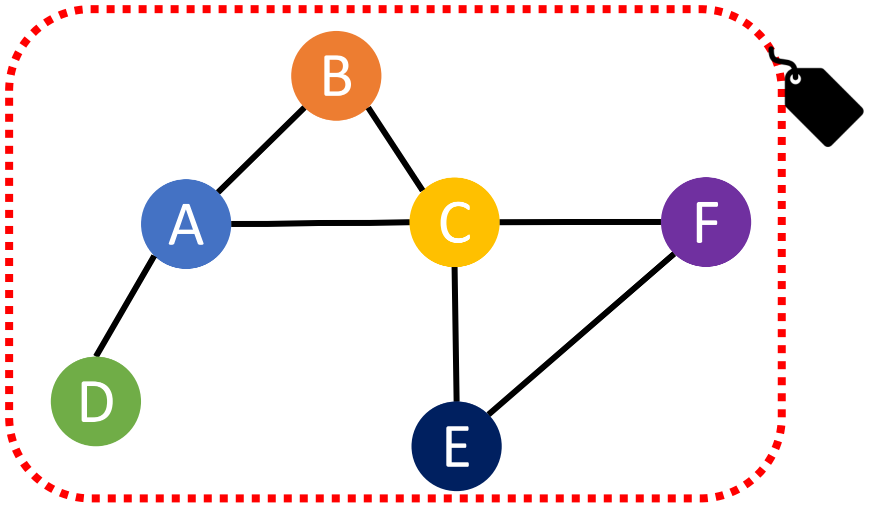
Downstream tasks

- Node-level prediction
- Edge-level prediction
- Attribute-level prediction



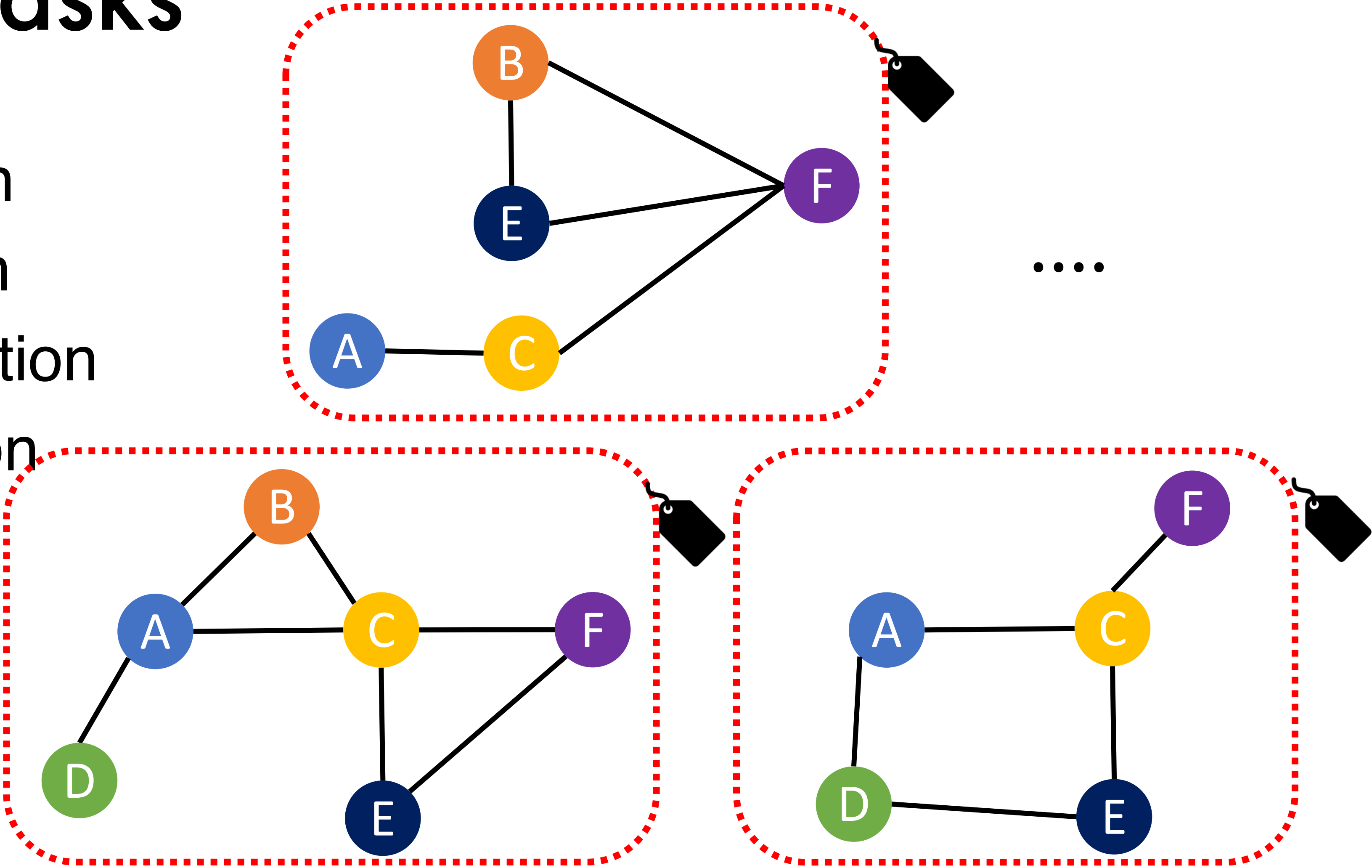
Downstream tasks

- Node-level prediction
- Edge-level prediction
- Attribute-level prediction
- Graph-level prediction



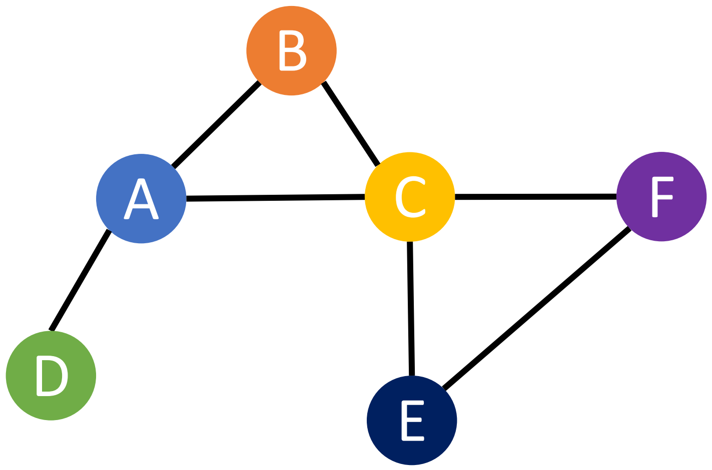
Downstream tasks

- Node-level prediction
- Edge-level prediction
- Attribute-level prediction
- Graph-level prediction

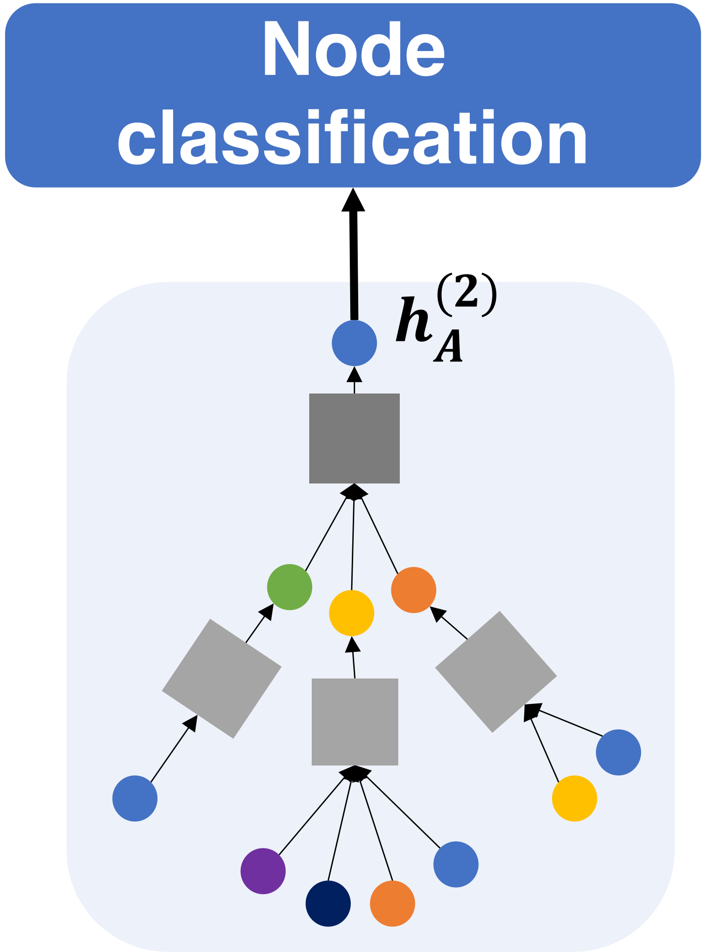


Downstream tasks

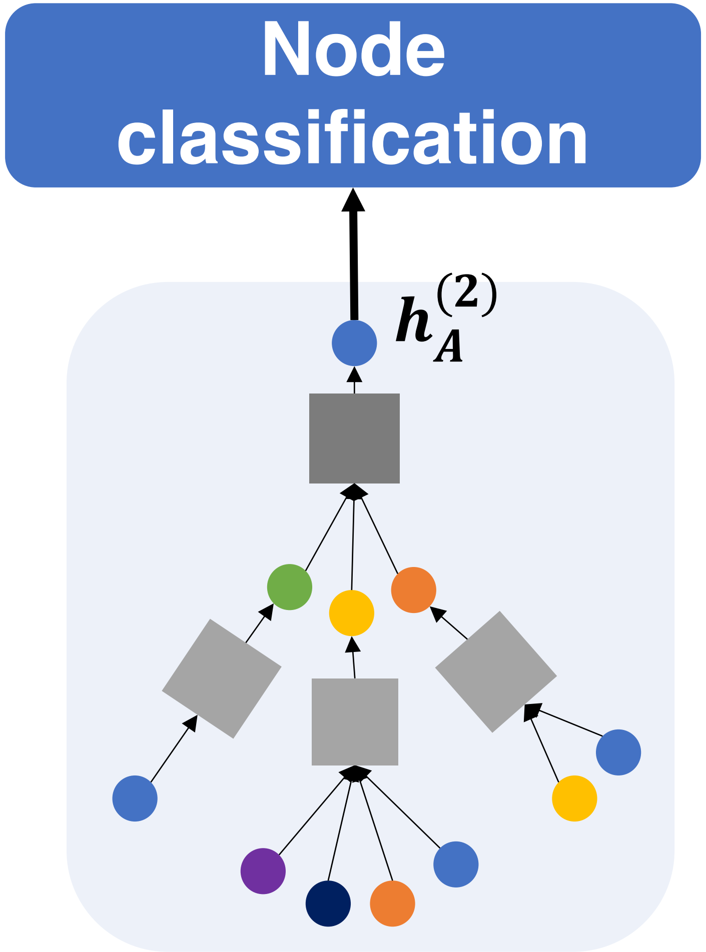
- **Node-level prediction**
- Edge-level prediction
- Attribute-level prediction
- **Graph-level prediction**

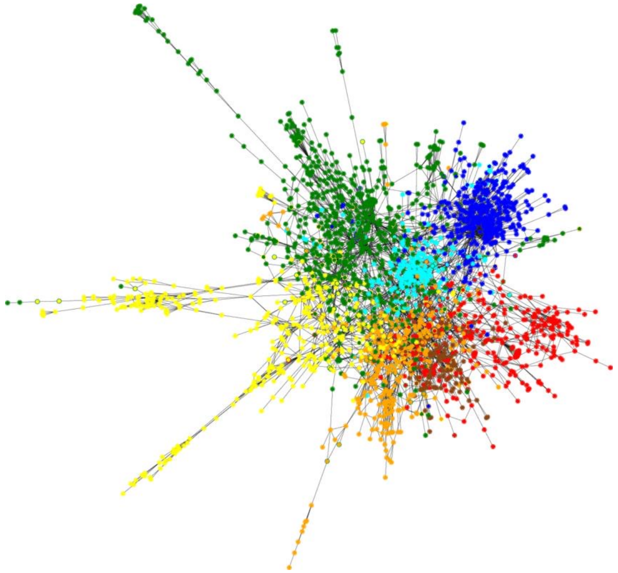


Node-level prediction tasks



Node-level prediction tasks





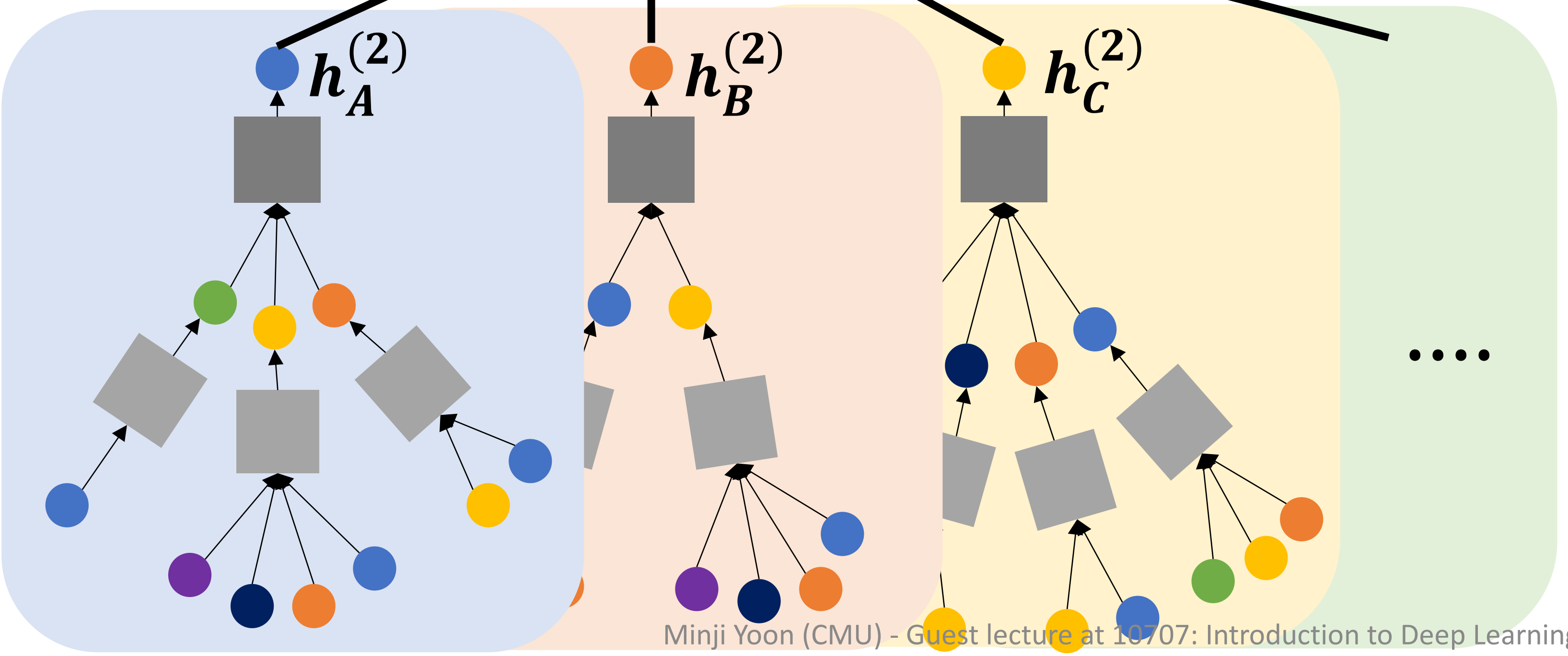
- Classify **papers** into topics on **citation networks**
- Cluster **posts** into subgroups on **Reddit networks**
- Classify **products** into categories on **Amazon co-purchase graphs**

Graph-level prediction tasks

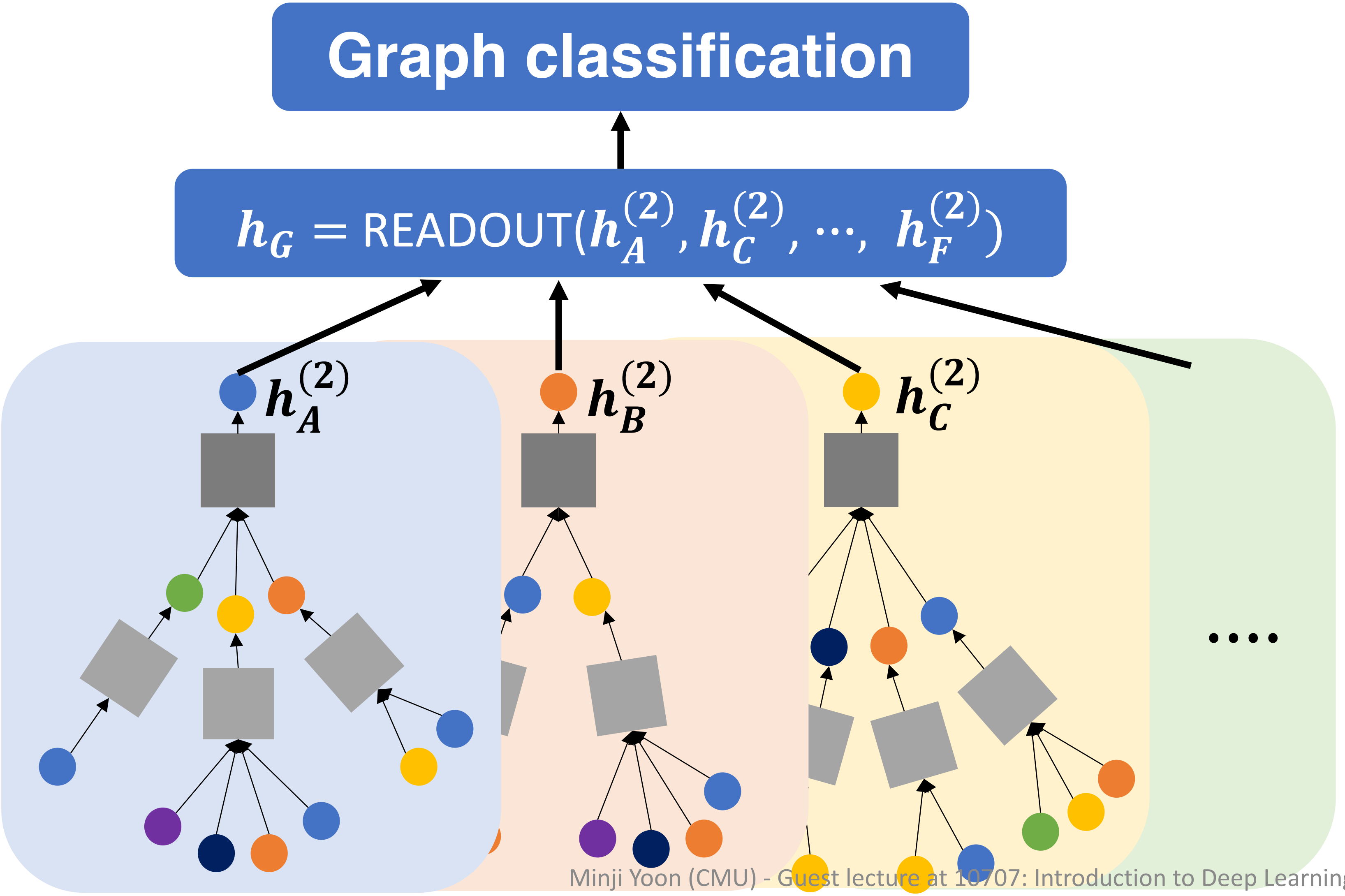
Graph classification

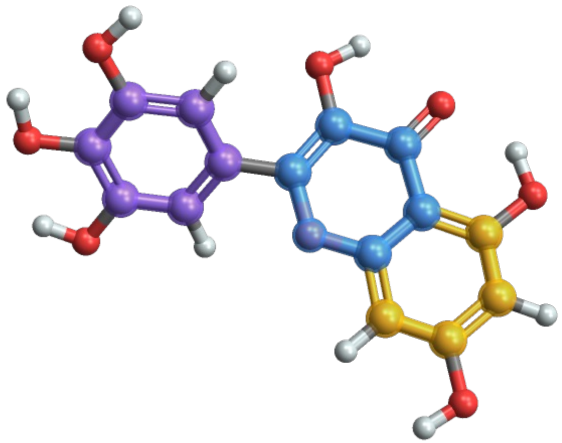
$$h_G = \text{READOUT}(h_A^{(2)}, h_C^{(2)}, \cdots, h_F^{(2)})$$

(ex) sum, average, min/max pooling of node embeddings



Graph-level prediction tasks





- Predict **properties of a molecule (graph)** where nodes are atoms and edges are chemical bonds

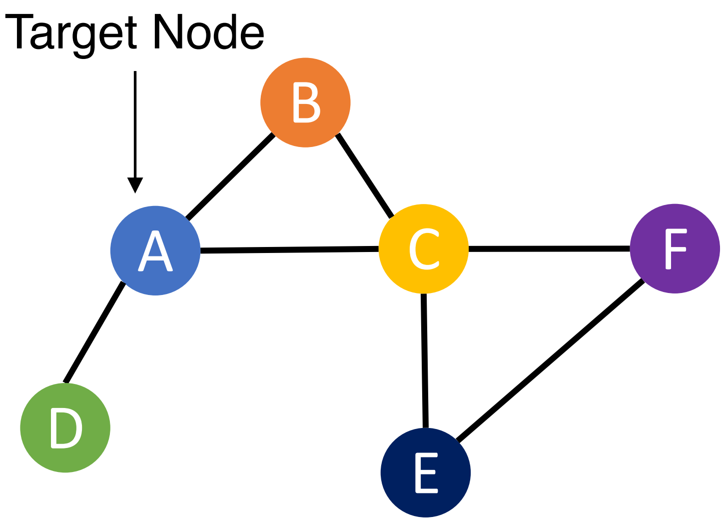
So far, we have talked about..

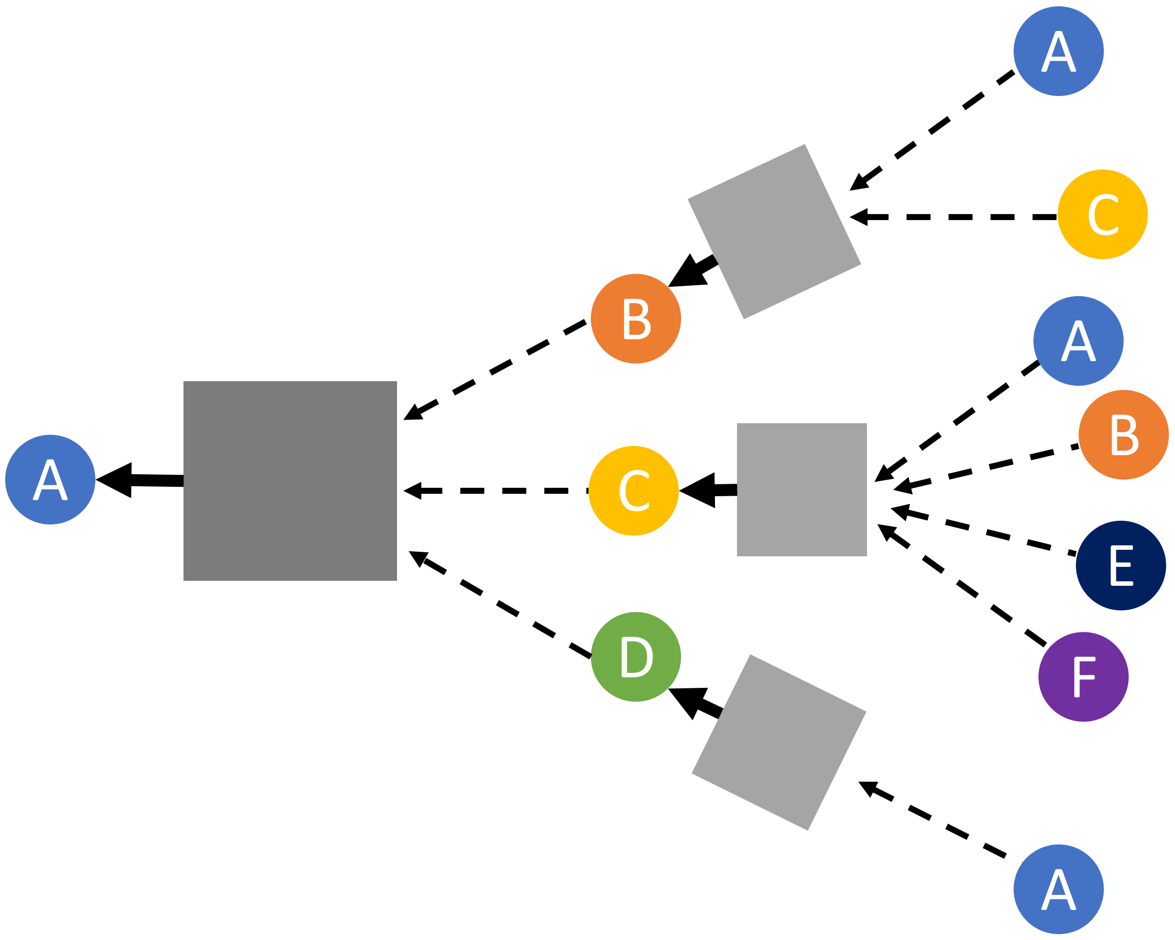
1. **Graph Neural Network**
   - Problem definition
   - Skeleton
     - Aggregation operation
     - Transformation operation
2. **Implementation**
   - Computation graph
   - Batch execution
3. **Downstream tasks**
   - Node-level prediction
   - Graph-level prediction

So far, we have talked about..

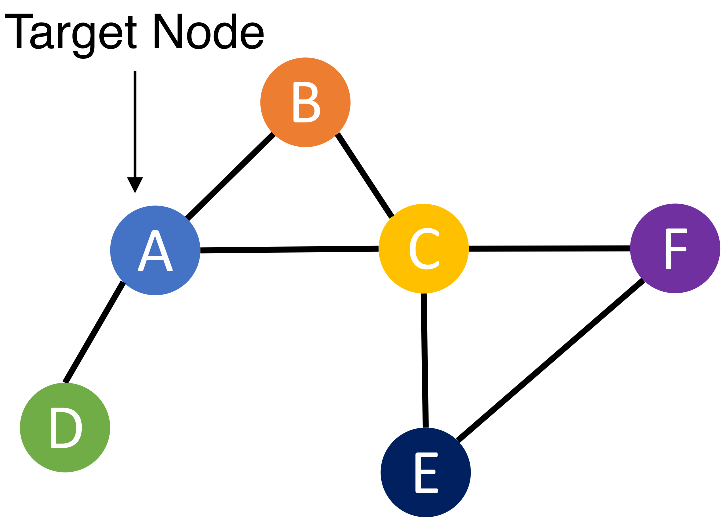
1. **Graph Neural Network**
    - Problem definition
    - Skeleton
        - Aggregation operation
        - Transformation operation
2. **Implementation**
    - Computation graph
    - Batch execution
3. **Downstream tasks**
    - Node-level prediction
    - Graph-level prediction

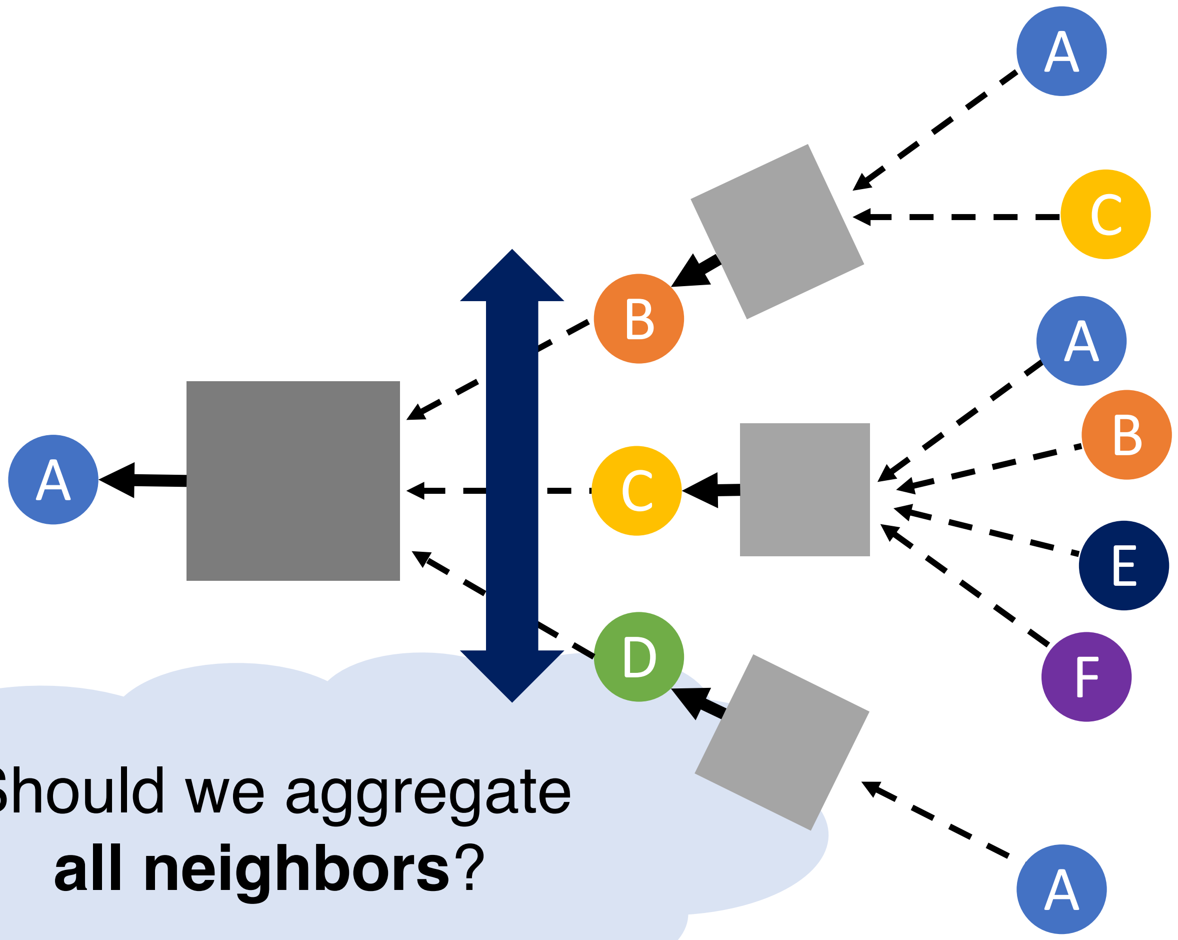
Graph Neural Networks





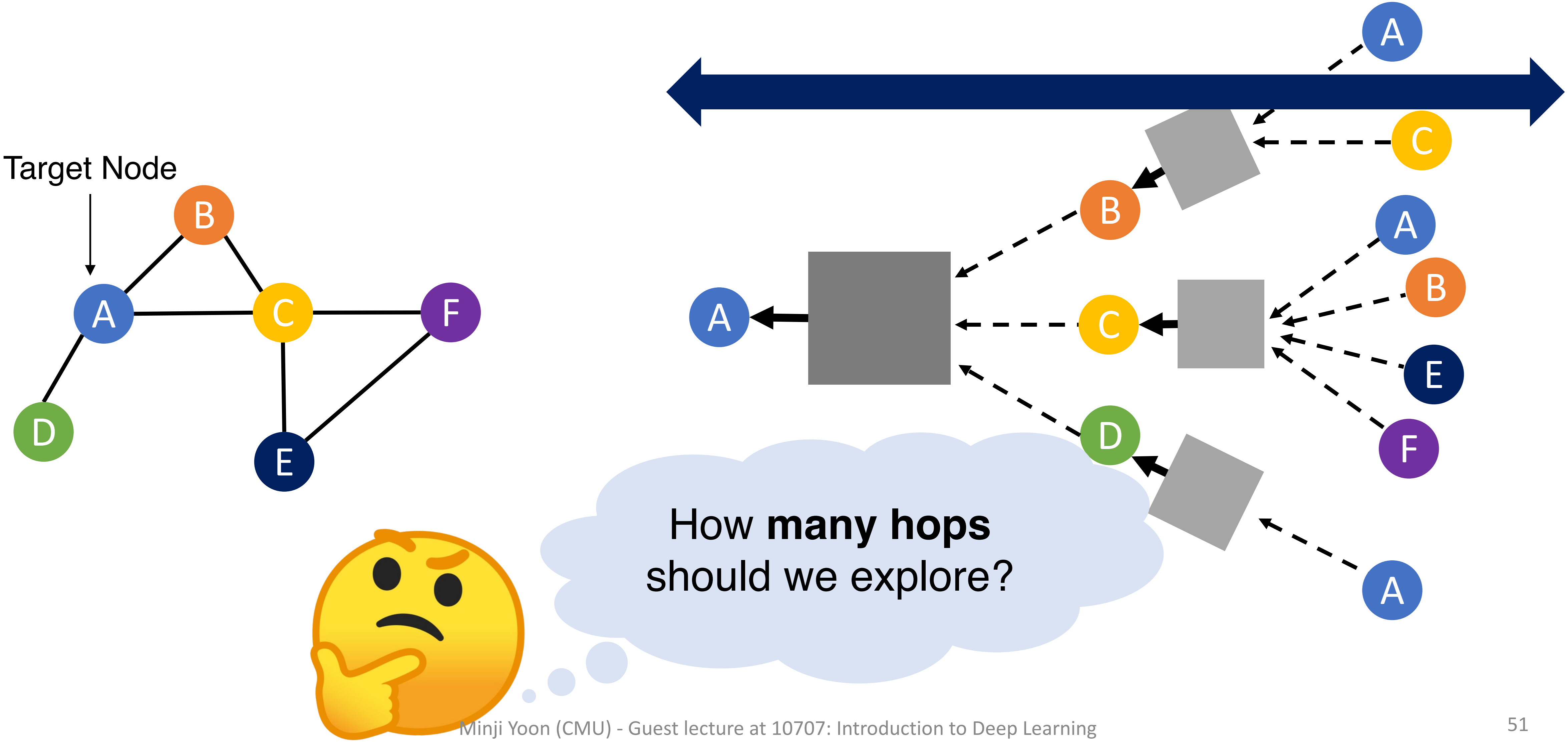
Graph Neural Networks - Width



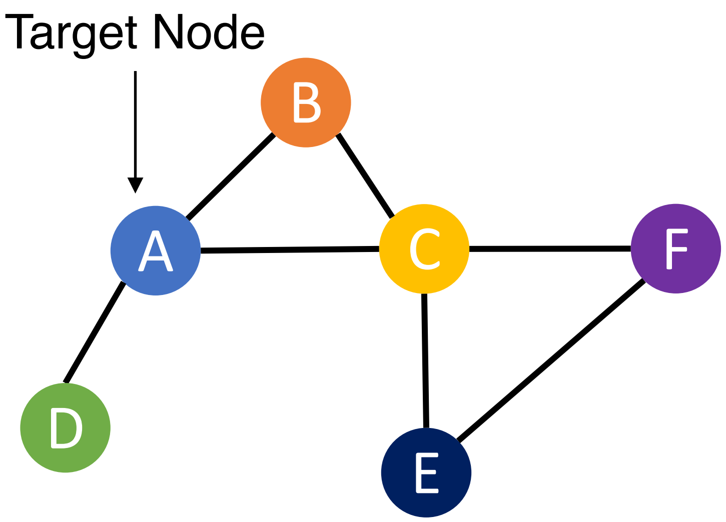


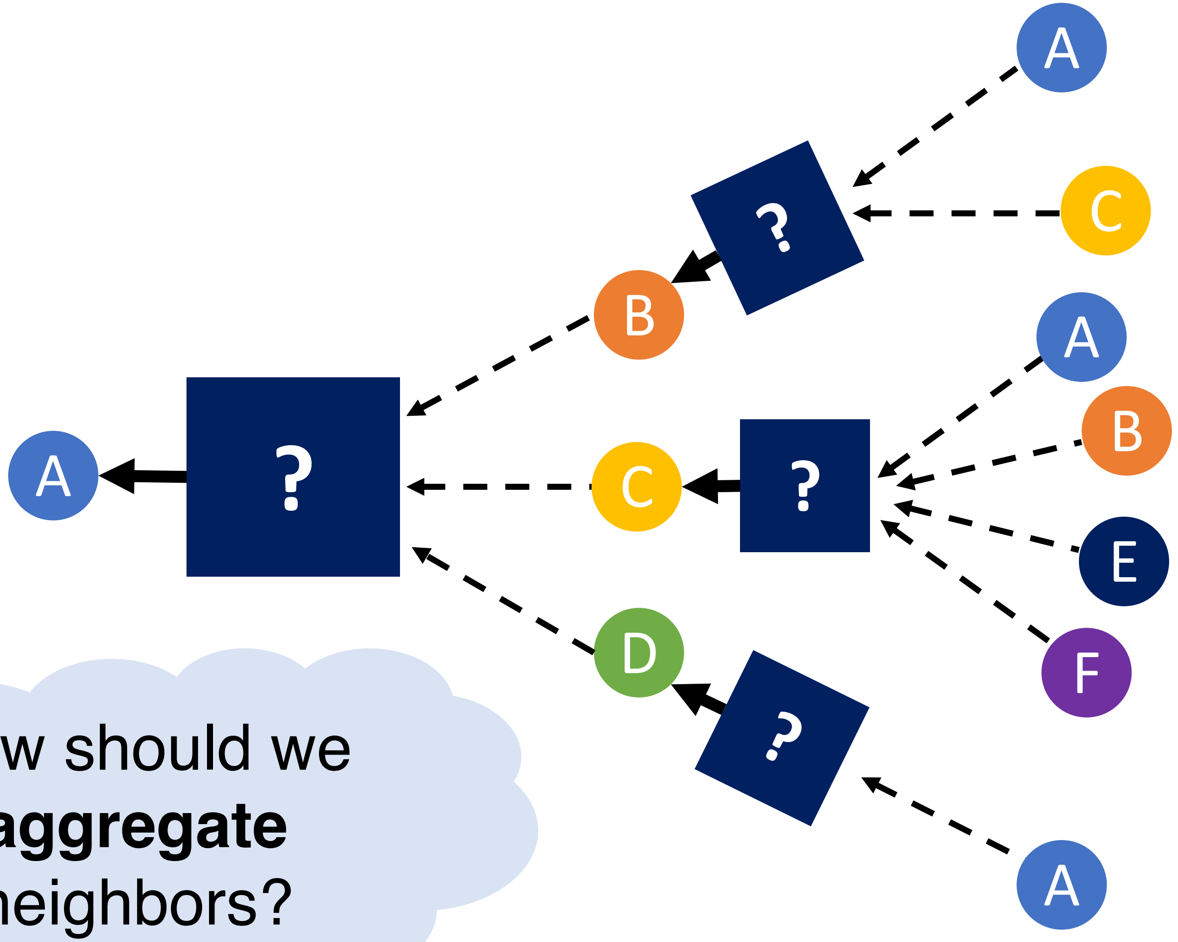
Should we aggregate **all neighbors**?

Graph Neural Networks - Depth



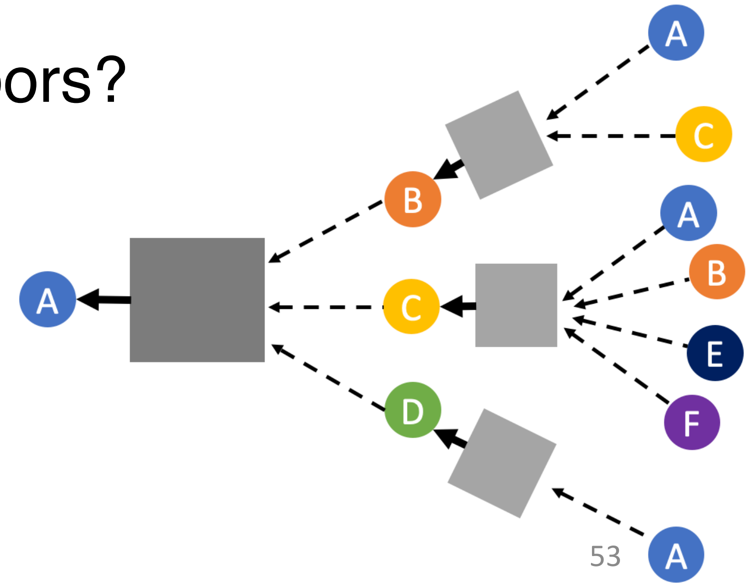
Graph Neural Networks - Aggregation





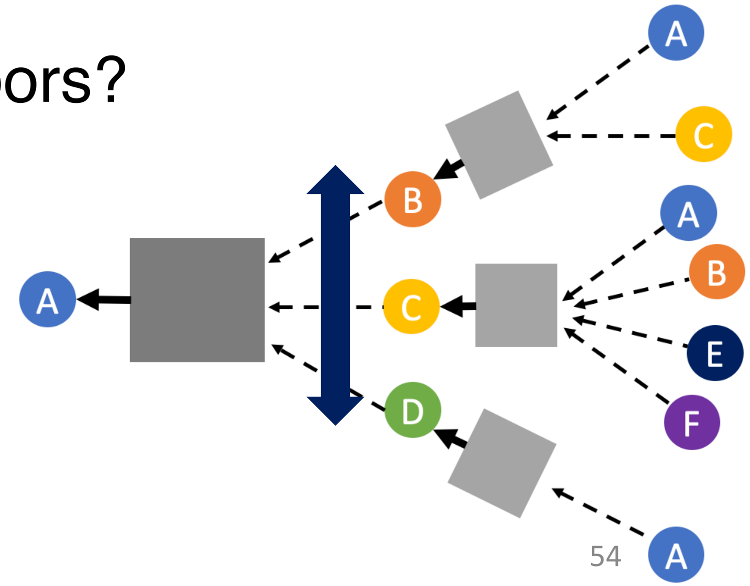
Graph Neural Network Architectures

- Width
  - Which neighbors should we aggregate messages from?
- Depth
  - How many hops should we check?
- Aggregation
  - How should we aggregate messages from neighbors?



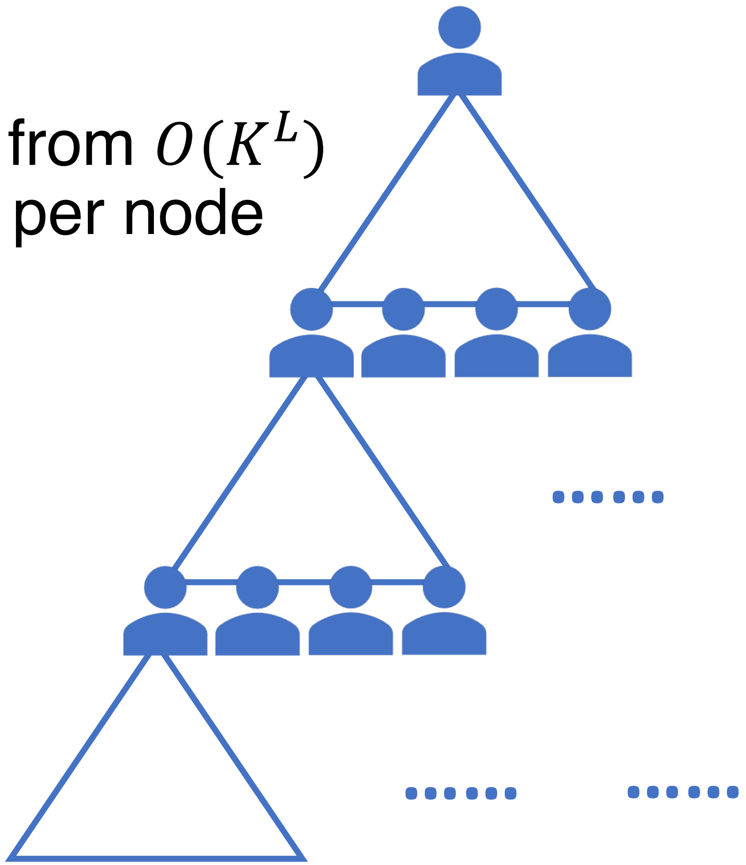
Graph Neural Network Architectures

- **Width**
  - **Which neighbors should we aggregate messages from?**
- Depth
  - How many hops should we check?
- Aggregation
  - How should we aggregate messages from neighbors?



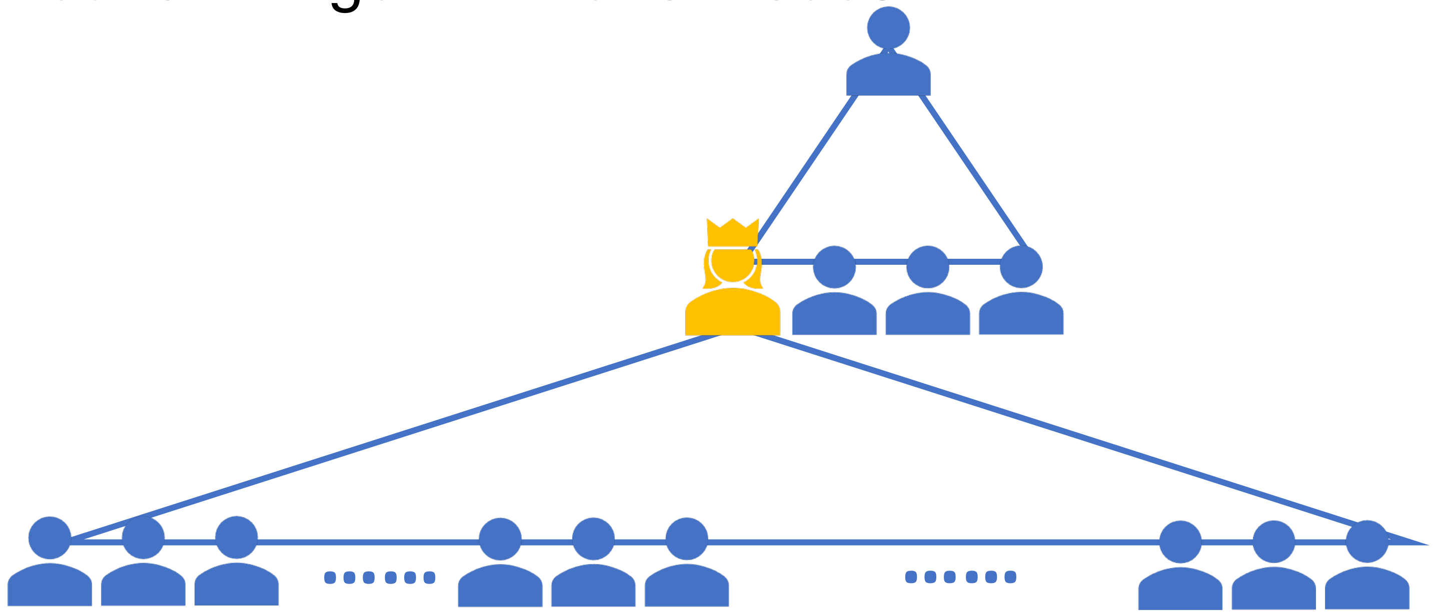
Aggregation Width in GNNs

- If we aggregate all neighbors, GNNs have scalability issues
- Neighbor explosion
  - In L -layer GNNs, one node aggregates information from $O(K^L)$ nodes where K is the average number of neighbors per node



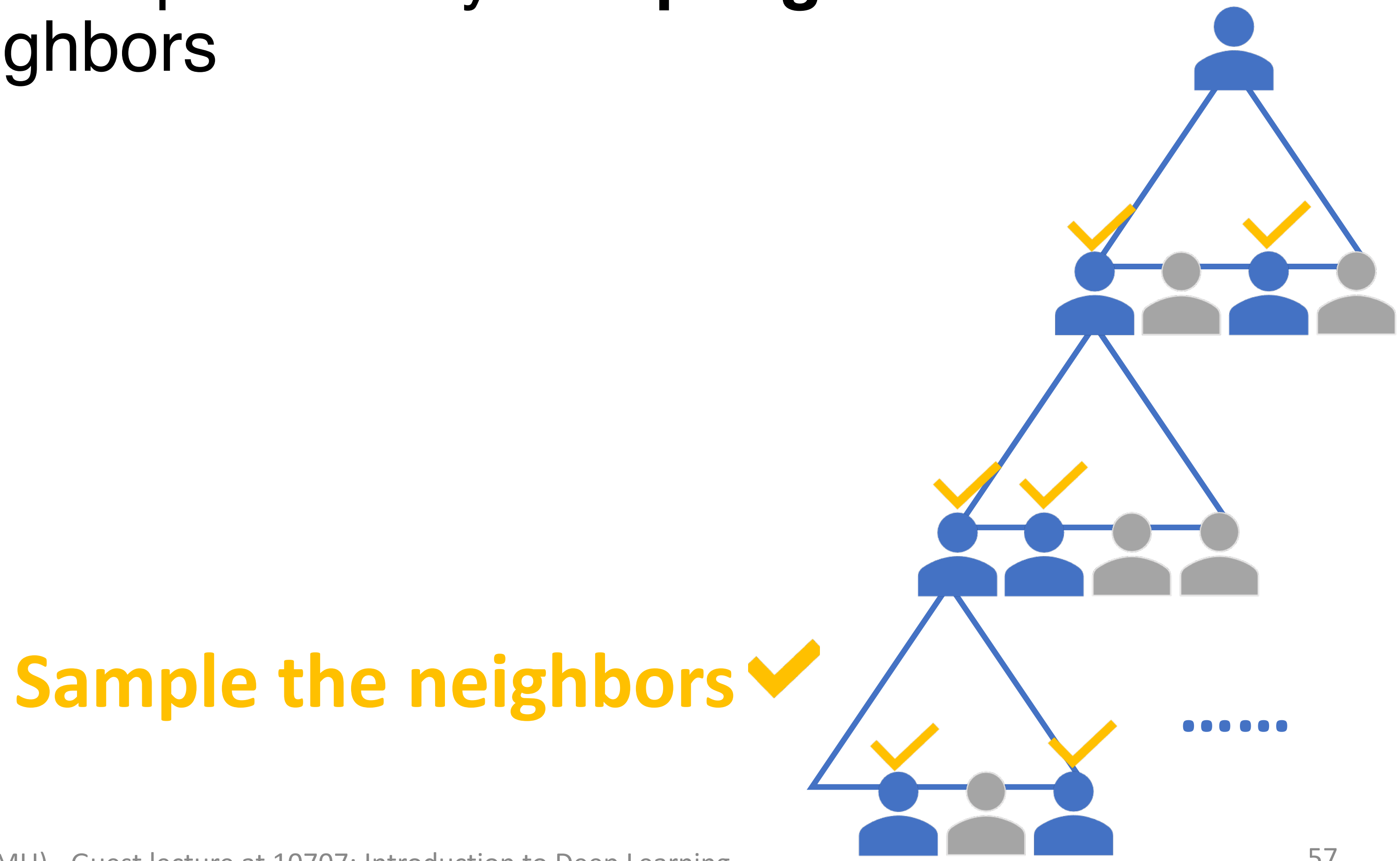
Aggregation Width in GNNs

- If we aggregate all neighbors, GNNs have scalability issues
- Neighbor explosion
  - Hub nodes who are connected to a huge number of nodes



Aggregation Width in GNNs

- Limit the neighborhood expansion by **sampling** a fixed number of neighbors



Aggregation Width in GNNs

- Random sampling
  - Assign **same** sampling probabilities to all neighbors
  - *GraphSage*[4]
- Importance sampling
  - Assign **different** sampling probabilities to all neighbors
  - *FastGCN*[5], *LADIES*[6], *AS-GCN*[7], *GCN-BS*[8], *PASS*[9]

[4] Will Hamilton, et al. "Inductive representation learning on large graphs"
[5] Jie Chen, et al. "Fastgcn: fast learning with graph convolutional networks via importance sampling"
[6] Difan Zou, et al. "Layer-Dependent Importance Sampling for Training Deep and Large Graph Convolutional Networks"
[7] Wenbing Huang, et al. "Adaptive sampling towards fast graph representation learning"
[8] Ziqi Liu, et al. "Bandit Samplers for Training Graph Neural Networks"
[9] Minji Yoon, et al. "Performance-Adaptive Sampling Strategy Towards Fast and Accurate Graph Neural Networks"

Aggregation Width in GNNs

Importance sampling

: assign *higher sampling probabilities to neighbors* who

- **Minimize variance in sampling**
  - *FastGCN*[5], *LADIES*[6], *AS-GCN*[7], *GCN-BS*[8]
- **Maximize GNN performance**
  - *PASS*[9]

[4] Will Hamilton, et al. “Inductive representation learning on large graphs”
[5] Jie Chen, et al. “Fastgcn: fast learning with graph convolutional networks via importance sampling”
[6] Difan Zou, et al. “Layer-Dependent Importance Sampling for Training Deep and Large Graph Convolutional Networks”
[7] Wenbing Huang, et al. “Adaptive sampling towards fast graph representation learning”
[8] Ziqi Liu, et al. “Bandit Samplers for Training Graph Neural Networks”
[9] Minji Yoon, et al. “Performance-Adaptive Sampling Strategy Towards Fast and Accurate Graph Neural Networks”

Aggregation Width in GNNs

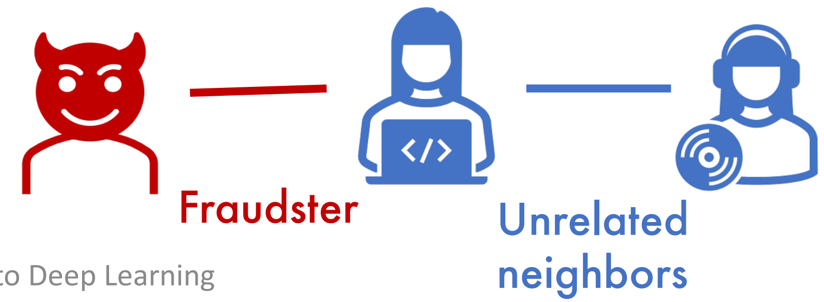
Method	Cora	Citeseer	Pubmed	AmazonC	AmazonP	MsCS	MsPhysics
FastGCN	0.582	0.496	0.569	0.480	0.542	0.520	0.638
AS-GCN	0.462	0.387	0.502	0.419	0.480	0.403	0.516
GraphSage	0.788	0.698	0.792	0.707	0.787	0.766	0.875
GCN-BS	0.788	0.693	0.809	0.736	0.800	0.780	0.887
PASS	**0.821**	**0.715**	**0.858**	**0.757**	**0.855**	**0.884**	**0.934**

- **Node classification task on 7 different real-world graphs**
- **PASS outperforms all variance-minimizing methods by up to 10.4%**

Aggregation Width in GNNs

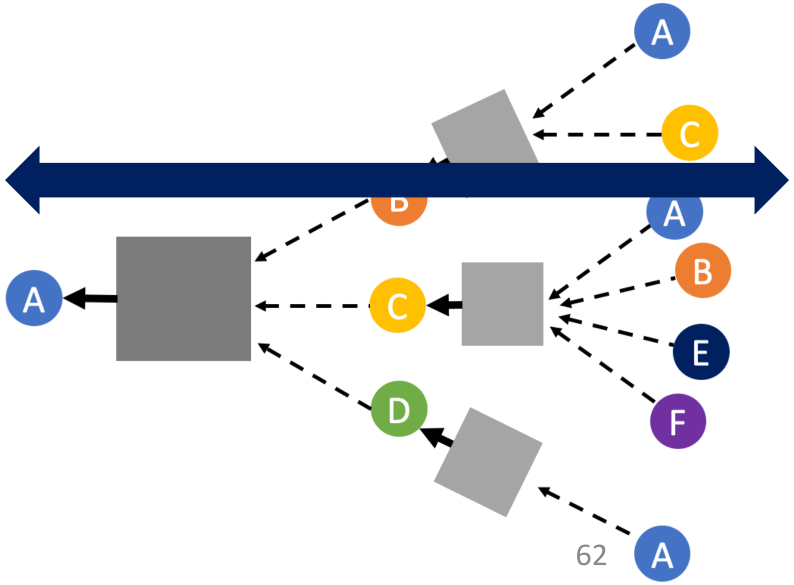
Method	Cora	Citeseer	Pubmed	AmazonC	AmazonP	MsCS	MsPhysics
FastGCN	0.582	0.496	0.569	0.480	0.542	0.520	0.638
AS-GCN	0.462	0.387	0.502	0.419	0.480	0.403	0.516
GraphSage	0.788	0.698	0.792	0.707	0.787	0.766	0.875
GCN-BS	0.788	0.693	0.809	0.736	0.800	0.780	0.887
PASS	**0.821**	**0.715**	**0.858**	**0.757**	**0.855**	**0.884**	**0.934**

Real-world graphs are noisy!!



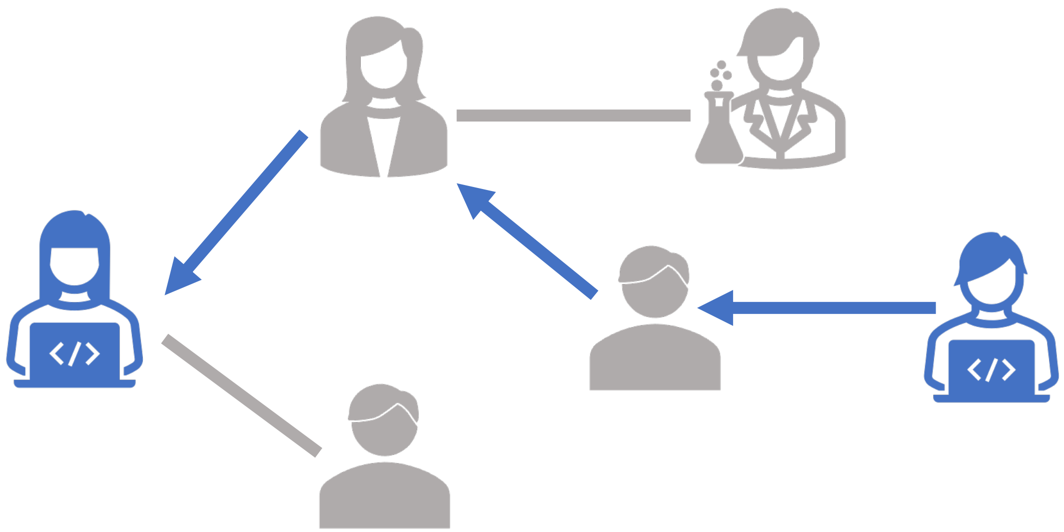
Graph Neural Network Architectures

- Width
  - Which neighbors should we aggregate messages from?
- **Depth**
  - **How many hops should we check?**
- Aggregation
  - How should we aggregate messages from neighbors?



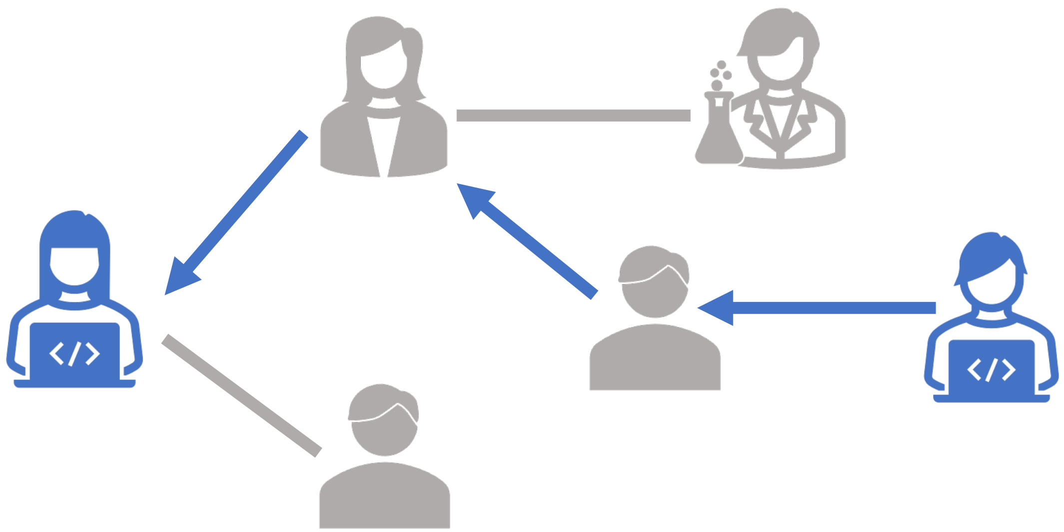
Aggregation Depth in GNNs

- Informative neighbors could be indirectly connected with a target node



Aggregation Depth in GNNs

- Informative neighbors could be indirectly connected with a target node
- Can’t we just look multiple hops away from the target node?



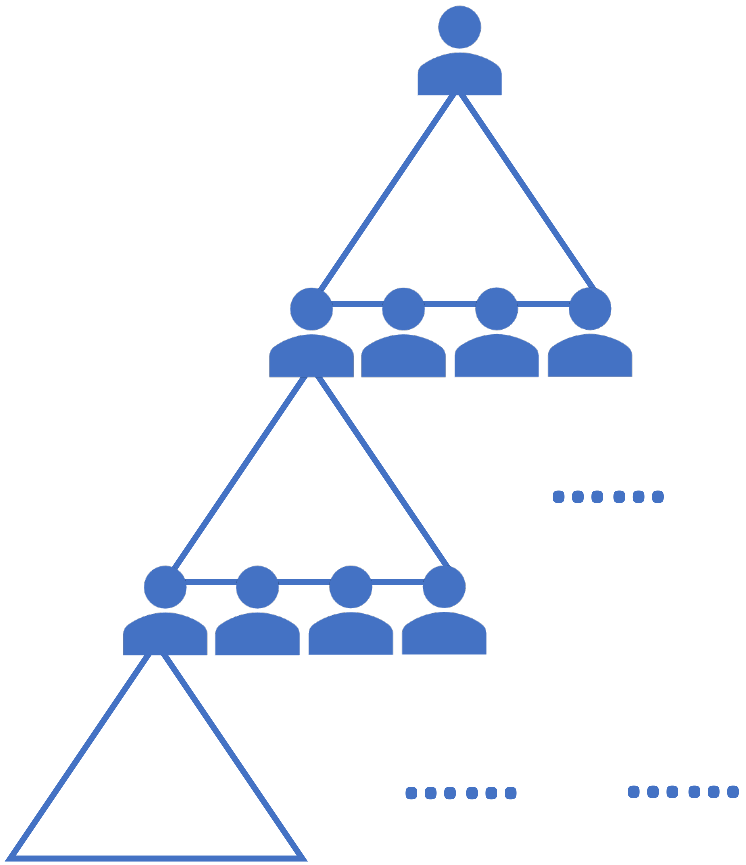
Aggregation Depth in GNNs

- 2-layer or 3-layer GNNs are commonly used in real worlds

Wasn’t it Deeeep Learning?

Aggregation Depth in GNNs

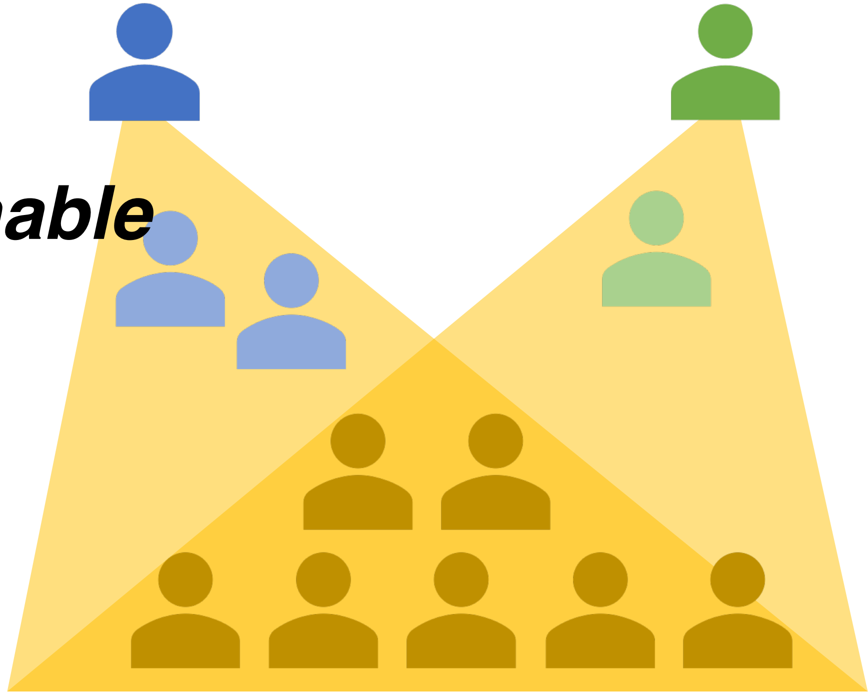
- When we increase the depth L more than this, GNNs face neighbor explosion $O(K^L)$
  - **Over-smoothing**
  - **Over-squashing**



Aggregation Depth in GNNs

Over-smoothing[10]

- When GNNs become deep, nodes share many neighbors
- Node embeddings become ***indistinguishable***

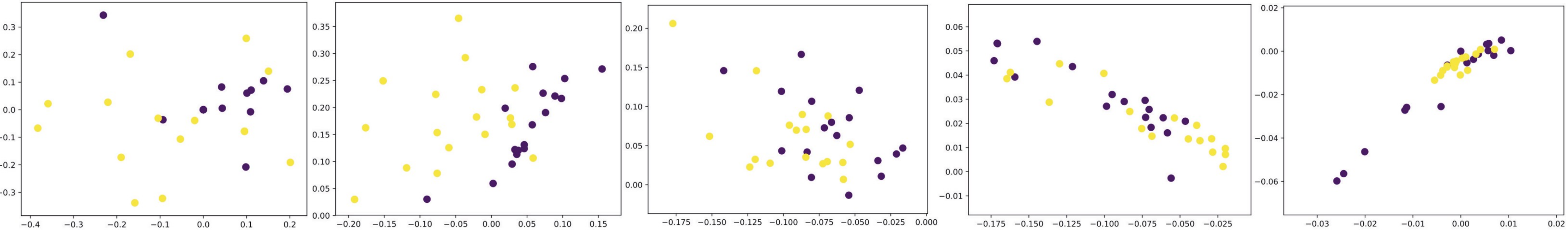


[10] Qimai Li, et al. "Deeper Insights into Graph Convolutional Networks for Semi-Supervised Learning"

Aggregation Depth in GNNs

Over-smoothing[10]

- Node embeddings of Zachary's karate club network with GNNs



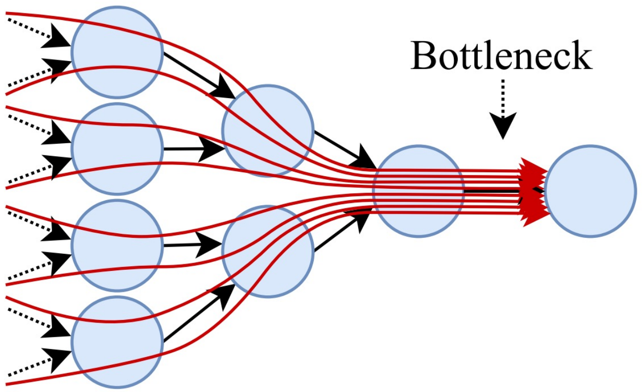
(a) 1-layer (b) 2-layer (c) 3-layer (d) 4-layer (e) 5-layer

[10] Qimai Li, et al. "Deeper Insights into Graph Convolutional Networks for Semi-Supervised Learning"

Aggregation Depth in GNNs

Over-squashing[12]

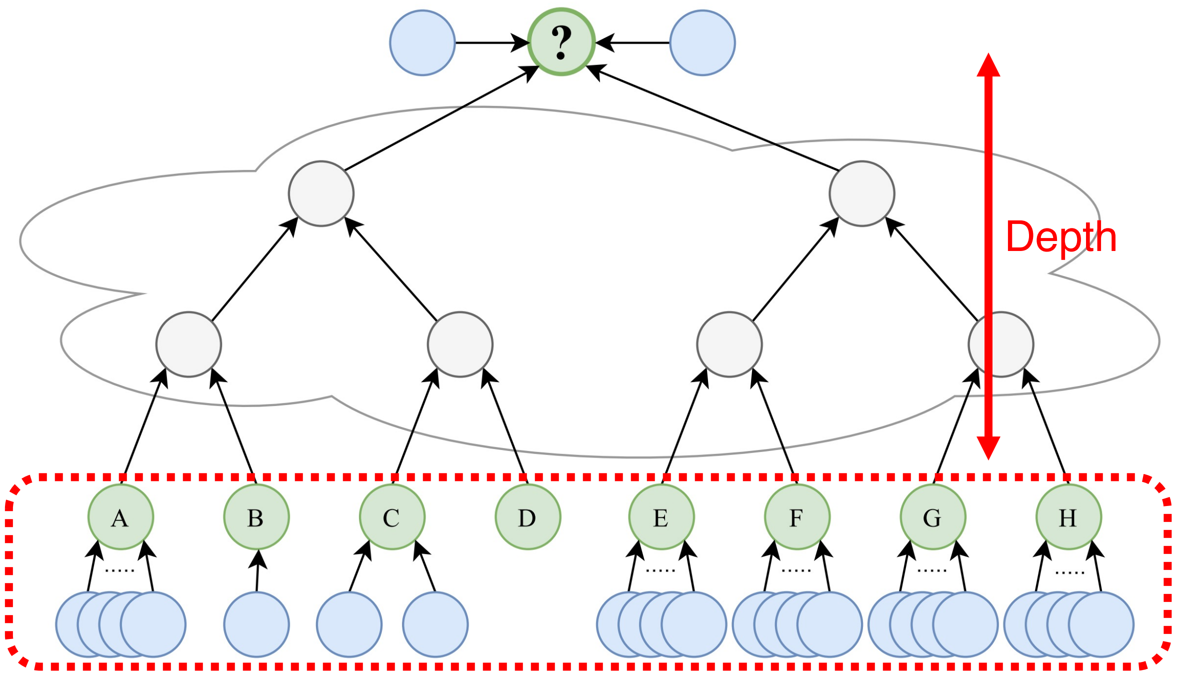
- A node's exponentially-growing neighborhood is compressed into a fixed-size vector

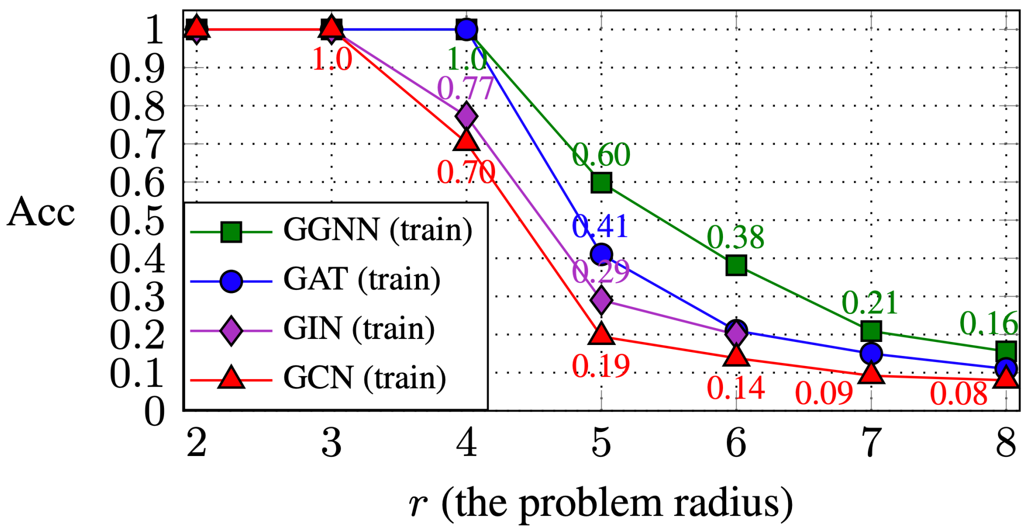


[12] Uri Alon, et al. "ON THE BOTTLENECK OF GRAPH NEURAL NETWORKS AND ITS PRACTICAL IMPLICATIONS"

Aggregation Depth in GNNs

Over-squashing[12]





[12] Uri Alon, et al. “ON THE BOTTLENECK OF GRAPH NEURAL NETWORKS AND ITS PRACTICAL IMPLICATIONS”

Aggregation Depth in GNNs

Decoupling the two concepts of depths in GNNs[13]

- **Depth-1**: neighborhood that each node aggregates information from
- **Depth-2**: number of layers in GNNs

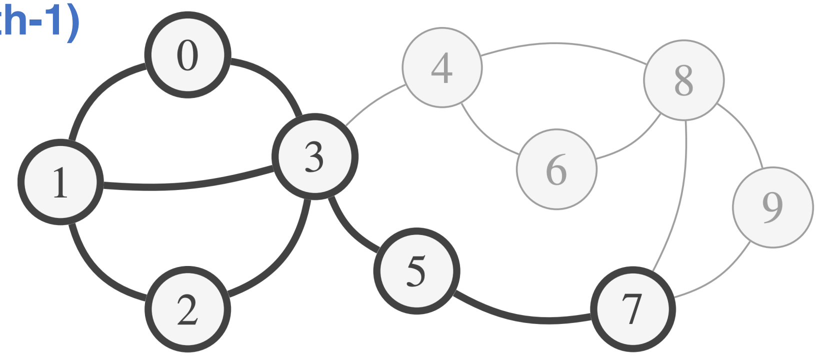
[13] Hanqing Zeng, et al. "Decoupling the Depth and Scope of Graph Neural Networks"

Aggregation Depth in GNNs

Decoupling the two concepts of depths in GNNs[13]

- **Depth-1**: neighborhood that each node aggregates information from
- **Depth-2**: number of layers in GNNs

Depth of neighborhood
(Depth-1)



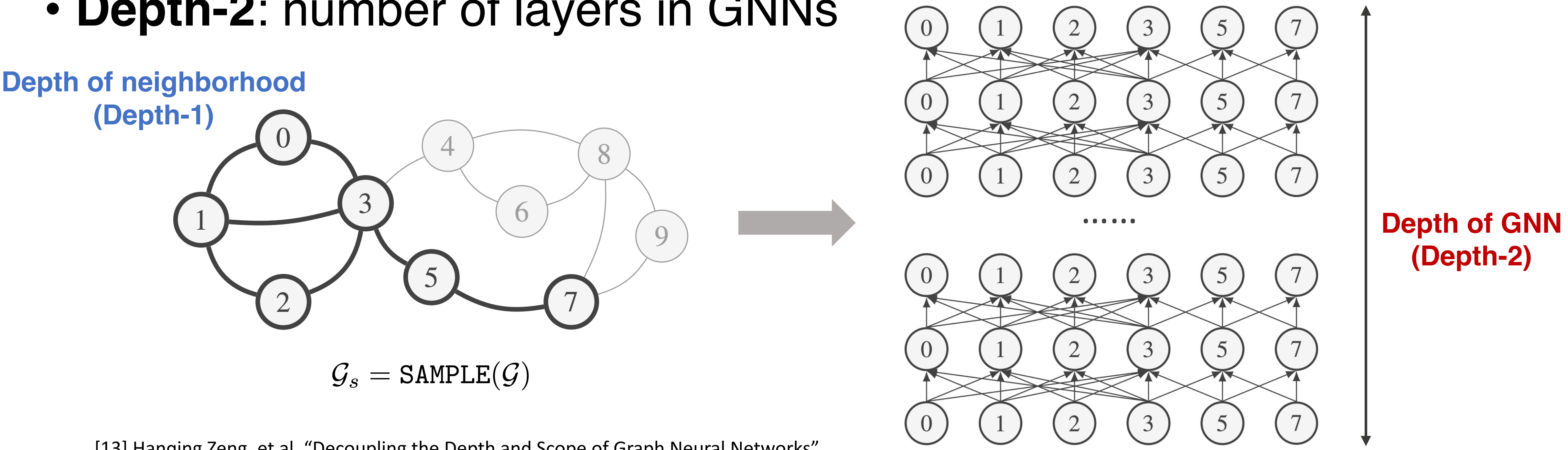
$$\mathcal{G}_s = \texttt{SAMPLE}(\mathcal{G})$$

[13] Hanqing Zeng, et al. "Decoupling the Depth and Scope of Graph Neural Networks"

Aggregation Depth in GNNs

Decoupling the two concepts of depths in GNNs[13]

- **Depth-1**: neighborhood that each node aggregates information from
- **Depth-2**: number of layers in GNNs

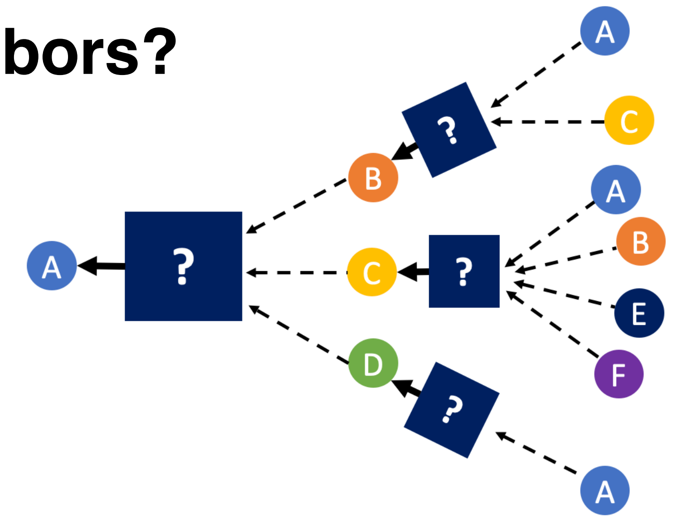


$\mathcal{G}_s = \texttt{SAMPLE}(\mathcal{G})$

[13] Hanqing Zeng, et al. "Decoupling the Depth and Scope of Graph Neural Networks"

Graph Neural Network Architectures

- Width
  - Which neighbors should we aggregate messages from?
- Depth
  - How many hops should we check?
- **Aggregation**
  - **How should we aggregate messages from neighbors?**



Aggregation strategy in GNNs

In each layer l :

Aggregate over neighbors

$$m_v^{(l-1)} = \boxed{\boldsymbol{f}^{(l)}}\left(h_v^{(l-1)}, \left\{h_u^{(l-1)} : u \in \mathcal{N}(v)\right\}\right)$$

Transform messages

$$h_v^{(l)} = \boldsymbol{g}^{(l)}(m_v^{(l-1)})$$

Aggregation strategy in GNNs

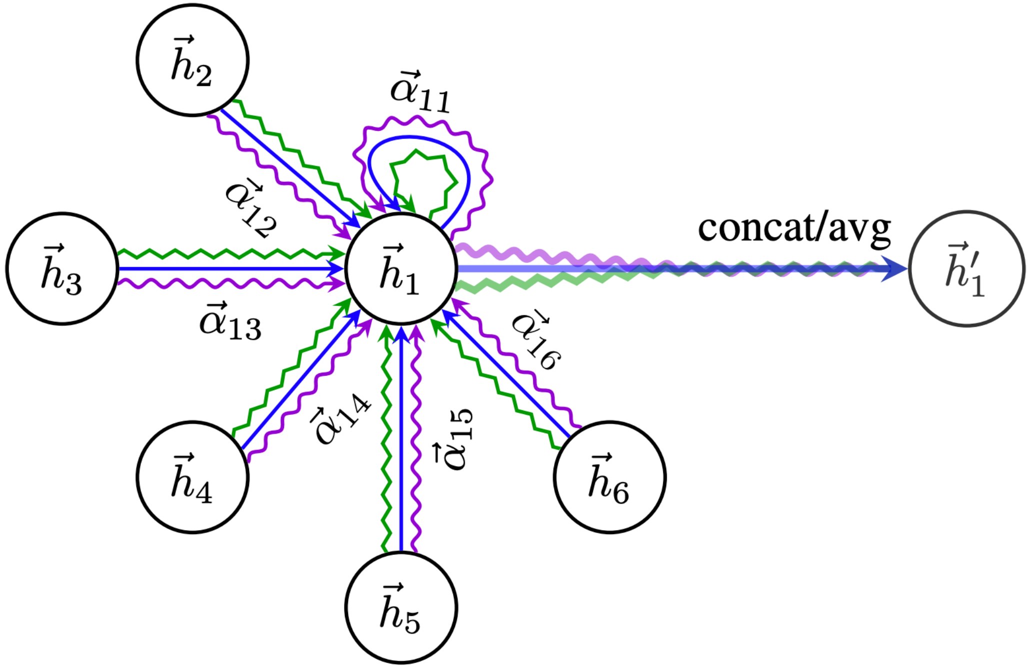
- GCN[1]
  - Average embeddings of neighboring nodes

[1] Kipf, Thomas N., et al. "Semi-supervised classification with graph convolutional networks."

Aggregation strategy in GNNs

- GAT[14]
  - Different weights to different nodes in a neighborhood
  - Multi-head attention

$$\alpha_{ij} = \frac{\exp\left(\text{LeakyReLU}\left(\vec{\mathbf{a}}^T[\mathbf{W}\vec{h}_i \| \mathbf{W}\vec{h}_j]\right)\right)}{\sum_{k\in\mathcal{N}_i}\exp\left(\text{LeakyReLU}\left(\vec{\mathbf{a}}^T[\mathbf{W}\vec{h}_i \| \mathbf{W}\vec{h}_k]\right)\right)}$$



[14] Petar Veličković., et al. "GRAPH ATTENTION NETWORKS."

Aggregation strategy in GNNs

In each layer l :

Aggregate over neighbors

$$m_v^{(l-1)} = \boxed{\boldsymbol{f}^{(l)}}\left(h_v^{(l-1)}, \left\{h_u^{(l-1)} : u \in \mathcal{N}(v)\right\}\right)$$

Core part of GNNs

Transform messages

$$h_v^{(l)} = \boxed{\boldsymbol{g}^{(l)}}(m_v^{(l-1)})$$

Any neural network module can fit in.
1-layer MLP is commonly used.

Aggregation strategy in GNNs

Power of **GNNs**

=

Power of **aggregation strategies**

Aggregation strategy in GNNs

- By measuring the power of GNNs, we can find the best aggregation strategy!!

Aggregation strategy in GNNs

- By measuring the expressive power of GNNs, we can find the best aggregation strategy!!
- *But.. what is the power of GNNs and how can we measure it?*

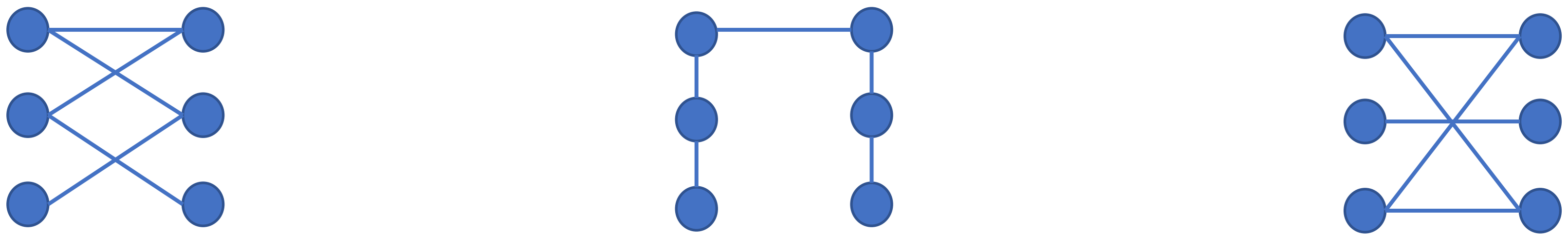
Aggregation strategy in GNNs

- How powerful are Graph Neural Networks?[2]
- Metric
  - Graph-level prediction task
  - Can a GNN model distinguish two non-isomorphic graphs?

[2] Keyulu Xu., et al. "HOW POWERFUL ARE GRAPH NEURAL NETWORKS?”

Aggregation strategy in GNNs

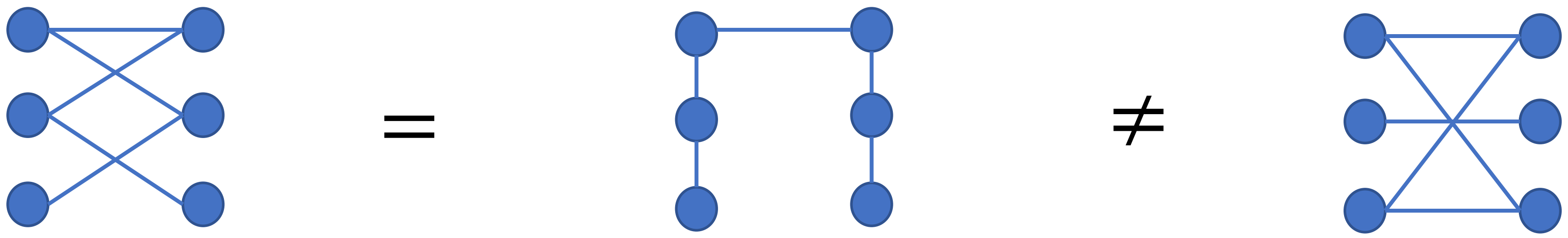
- How powerful are Graph Neural Networks?[2]
- Metric
  - Graph-level prediction task
  - Can a GNN model distinguish two non-isomorphic graphs?



[2] Keyulu Xu., et al. "HOW POWERFUL ARE GRAPH NEURAL NETWORKS?"

Aggregation strategy in GNNs

- How powerful are Graph Neural Networks?[2]
- Metric
  - Graph-level prediction task
  - Can a GNN model distinguish two non-isomorphic graphs?



[2] Keyulu Xu., et al. "HOW POWERFUL ARE GRAPH NEURAL NETWORKS?"

Aggregation strategy in GNNs

- How powerful are Graph Neural Networks?[2]
  - Any aggregation-based GNN is at most as powerful as the **WL test**[15]
  - Maximum power = aggregation strategy is injective

$$f(x_1) = f(x_2) \Rightarrow x_1 = x_2$$

[2] Keyulu Xu., et al. "HOW POWERFUL ARE GRAPH NEURAL NETWORKS?"
[15] Boris Weisfeiler and AA Leman. "A reduction of a graph to a canonical form and an algebra arising during this reduction"

Aggregation strategy in GNNs

- How powerful are Graph Neural Networks?[2]
  - Any aggregation-based GNN is at most as powerful as the **WL test**[15]
  - Maximum power = aggregation strategy is injective
  - (ex) summation

Mean and Max both fail, while Sum can distinguish them!!

[2] Keyulu Xu., et al. "HOW POWERFUL ARE GRAPH NEURAL NETWORKS?"
[15] Boris Weisfeiler and AA Leman. "A reduction of a graph to a canonical form and an algebra arising during this reduction"

Aggregation strategy in GNNs

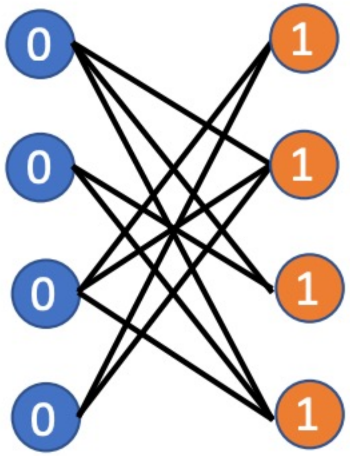
- Can we make more powerful GNNs?
    - Very active area, with many open problems

Aggregation strategy in GNNs

- Homophily assumption
  - Connected nodes are similar/related/informative

Aggregation strategy in GNNs

- Homophily assumption
  - Connected nodes are similar/related/informative
- How can we deal with **heterophilous networks**?[21,22]
  - Connected nodes have different class labels and dissimilar features



[21] Jiong Zhu., et al. "Beyond Homophily in Graph Neural Networks: Current Limitations and Effective Designs"
[22] Yao Ma, et al. "IS HOMOPHILY A NECESSITY FOR GRAPH NEURAL NETWORKS?"

Aggregation strategy in GNNs

Improved accuracy after filtering datasets

- Heterophilous graph datasets have serious drawbacks[23]

	squirrel		
	accuracy on original dataset	accuracy on filtered dataset	ranks
ResNet	33.88 ± 1.79	36.55 ± 1.82	12 / 7
ResNet+SGC	34.36 ± 1.21	38.36 ± 1.97	11 / 5
ResNet+adj	65.46 ± 1.58	38.37 ± 1.99	2 / 4
GCN	39.06 ± 1.52	39.47 ± 1.47	6 / 2
SAGE	35.83 ± 1.32	36.09 ± 1.99	9 / 9
GAT	32.21 ± 1.63	35.62 ± 2.06	14 / 11
GAT-sep	35.72 ± 1.98	35.46 ± 3.10	10 / 13
GT	31.61 ± 1.10	36.30 ± 1.98	15 / 8
GT-sep	36.08 ± 1.58	36.66 ± 1.63	8 / 6
H_2GCN	29.45 ± 1.65	35.10 ± 1.15	17 / 15
CPGNN	30.91 ± 1.98	30.04 ± 2.03	16 / 16
GPR-GNN	33.39 ± 2.05	38.95 ± 1.99	13 / 3
FSGNN	68.93 ± 1.69	35.92 ± 1.32	1 / 10
GloGNN	61.21 ± 1.96	35.11 ± 1.24	3 / 14
FAGCN	47.63 ± 1.85	41.08 ± 2.27	4 / 1
GBK-GNN	37.06 ± 1.24	35.51 ± 1.65	7 / 12
JacobiConv	46.17 ± 4.34	29.71 ± 1.66	5 / 17

[23] Oleg Platonov., et al. "A critical look at the evaluation of GNNs under heterophily: are we really making progress?"

So far, we have talked about..

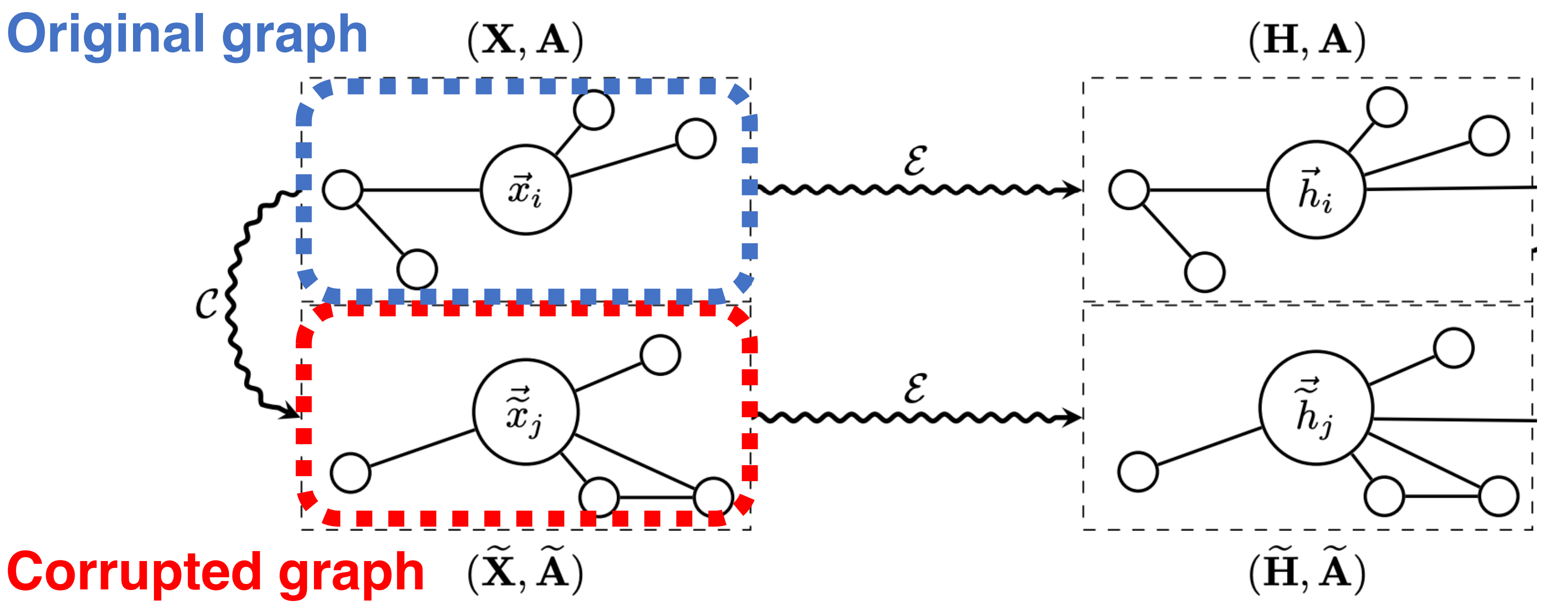
1. **Graph Neural Network**
   - Problem definition
   - Skeleton: aggregation, transformation operations
2. **Research questions in GNN architectures**
   - Width
   - Depth
   - Aggregation
3. **GNN training strategy**

How to train GNNs

- Semi-supervised learning
    - Input node features are given for all nodes in a graph
    - Only a subset of nodes have labels

How to train GNNs

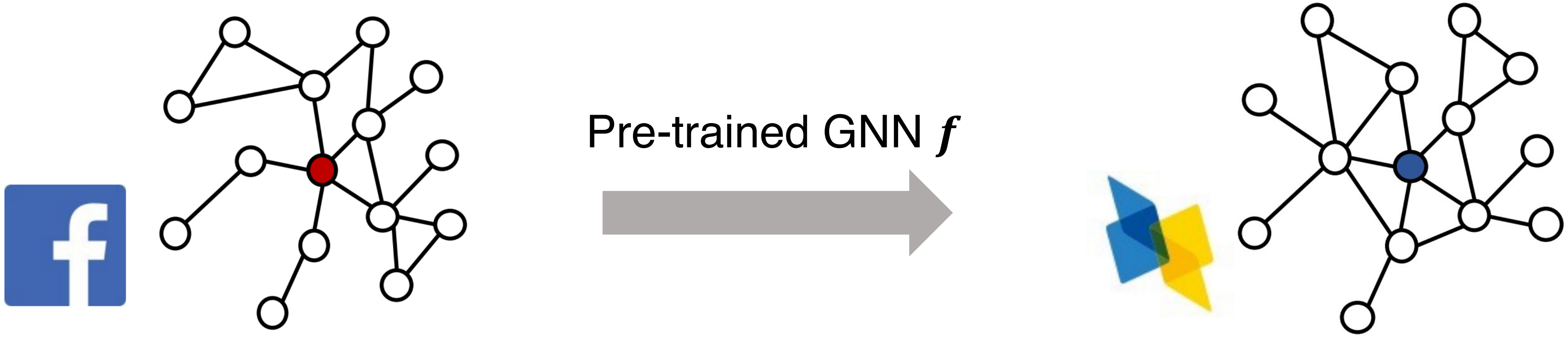
- Unsupervised learning[26]
  - Contrastive learning



[26] Petar Veličković., et al. "DEEP GRAPH INFOMAX"

How to train GNNs

- Transfer learning
  - Transfer a pre-trained GNN model between graphs[27]

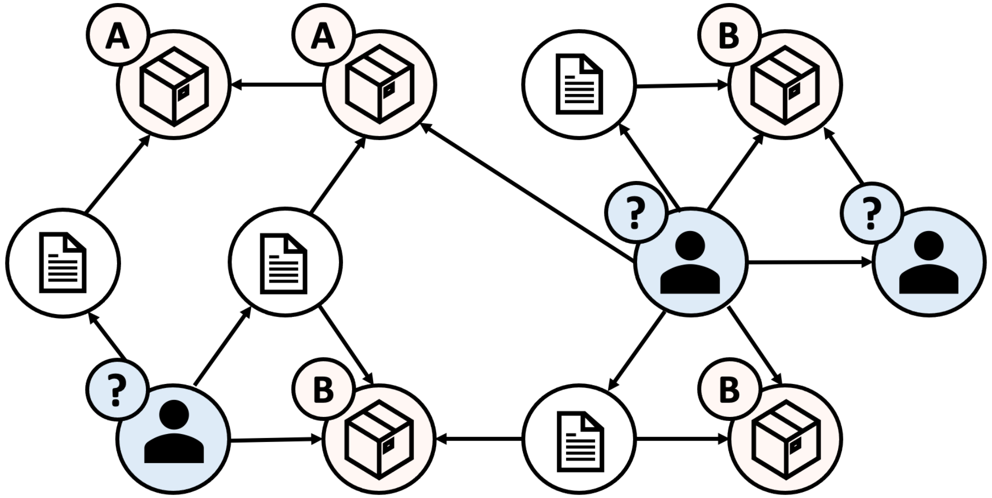


Facebook network **DBLP co-authorship network**

[27] Jiezhong Qiu, et al. "GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training"

How to train GNNs

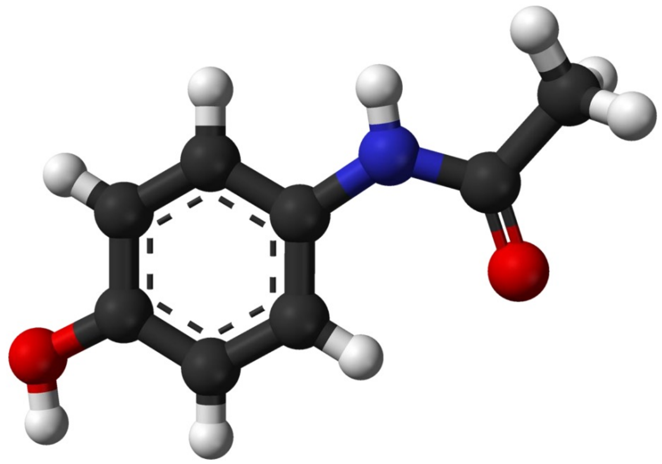
- Transfer learning
  - Transfer between different node types across a **heterogeneous graph**[28]

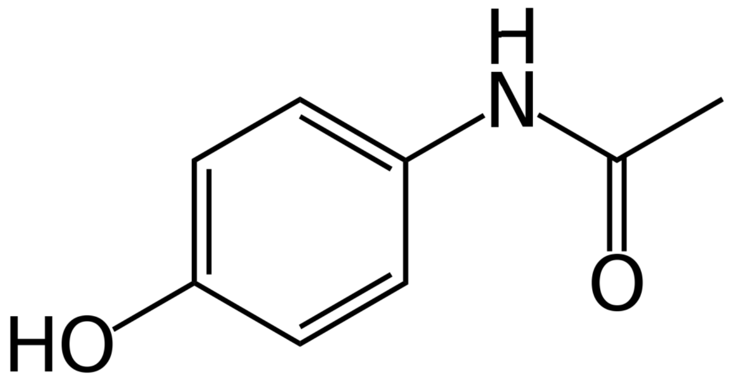


[28] Minji Yoon, et al. "Zero-shot Domain Adaptation of Heterogeneous Graphs via Knowledge Transfer Networks "

Impactful applications in science

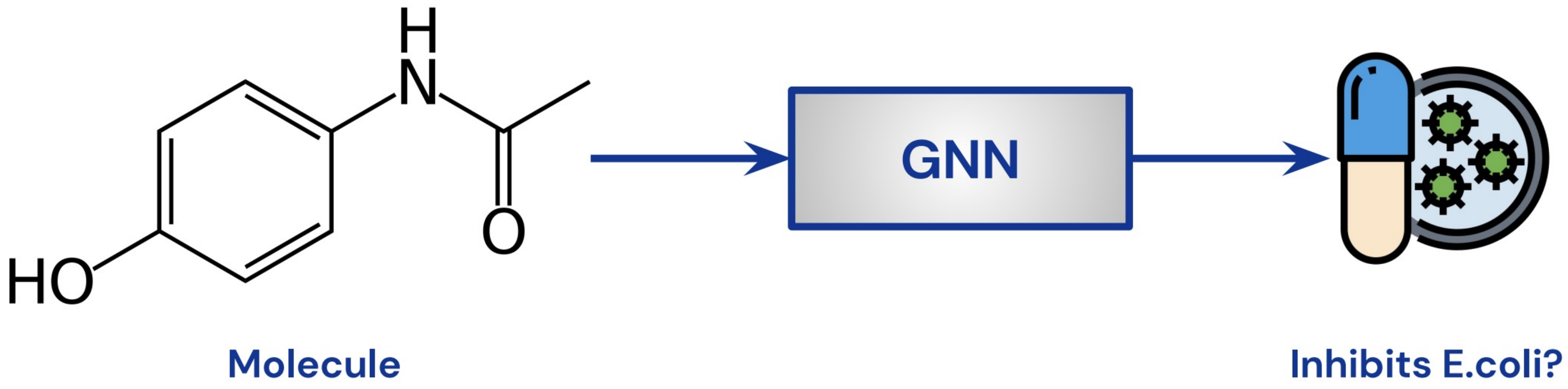
- GNNs for molecule classification
- Molecule
  - Node: atoms
  - Edge: bonds
  - Input features: atom type, charge, bond type





Impactful applications in science

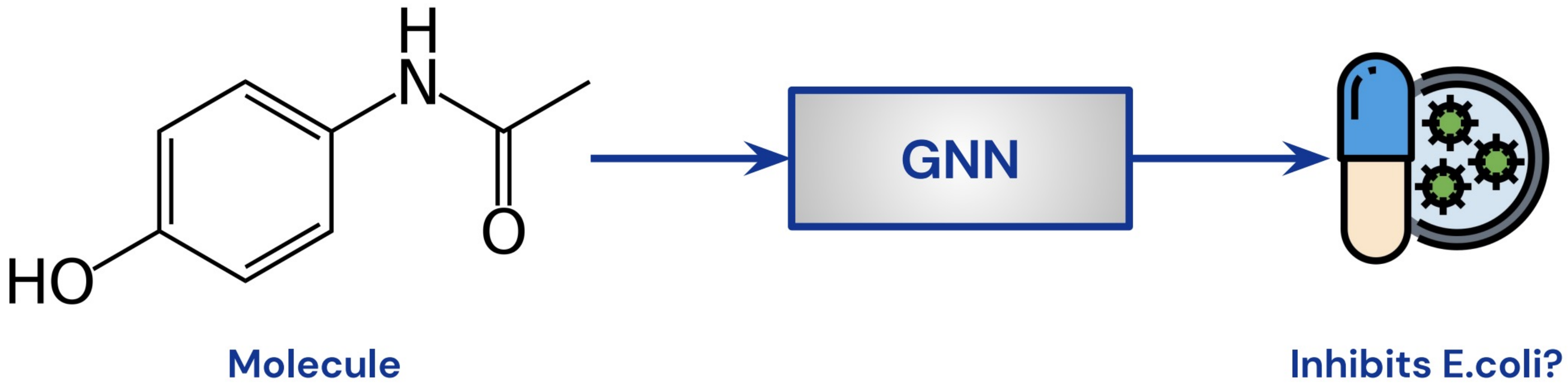
- Graph-level prediction: whether the molecule is a potent **drug**[29]
  - Binary classification on whether the drug will inhibit certain bacteria



[29] Jonathan M.Stokes, et al. "A Deep Learning Approach to Antibiotic Discovery”

Impactful applications in science

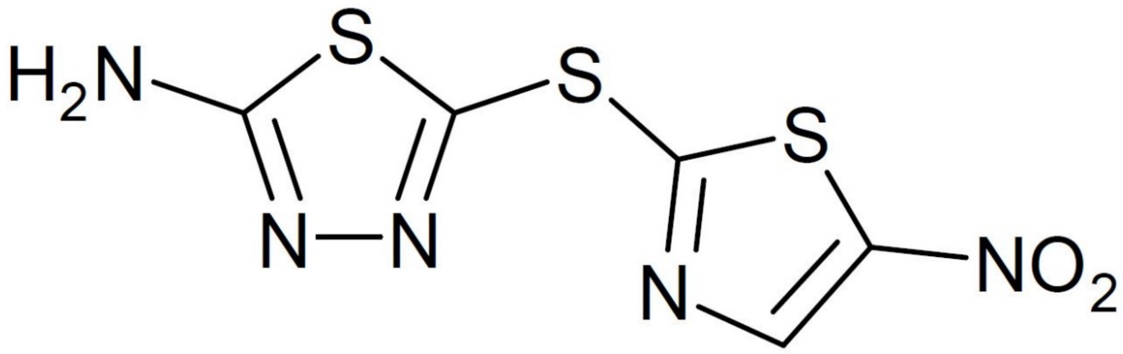
- Graph-level prediction: whether the molecule is a potent **drug**[29]
  - Execute on a large dataset of known candidate molecules
  - Select the ~ *top-100* candidates from the GNN model
  - Have chemists thoroughly investigate those



[29] Jonathan M.Stokes, et al. "A Deep Learning Approach to Antibiotic Discovery"

Impactful applications in science

- Discover a previously overlooked compound that is a **highly potent** antibiotic[29]



Halicin

[29] Jonathan M.Stokes, et al. "A Deep Learning Approach to Antibiotic Discovery”

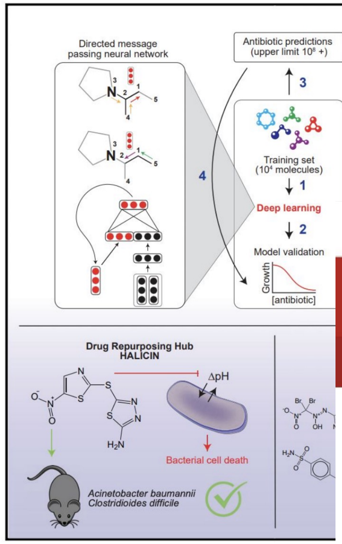
(slide: P. Veličković)

Impactful applications in science

Cell

A Deep Learning Approach to Antibiotic Discovery

Graphical Abstract



Authors

Jonathan M. Stokes, Kevin Yang, Kyle Swanson, ..., Tommi S. Jaakkola, Regina Barzilay, James J. Collins

Correspondence

regina@csail.mit.edu (R.B.), jimjc@mit.edu (J.J.C.)

nature Subscribe

NEWS · 20 FEBRUARY 2020

Powerful antibiotics discovered using AI

Machine learning spots molecules that work even against 'untreatable' strains of bacteria.

BBC Sign in | News | Sport | Reel | Worklife | Travel | Future

NEWS

Home | Video | World | UK | Business | Tech | Science | Stories | Entertainment & Arts

Scientists discover powerful antibiotic using AI

21 February 2020 Share

FINANCIAL TIMES

COMPANIES TECH MARKETS GRAPHICS OPINION WORK & CAREERS LIFE & ARTS HOW TO SPEND IT

intelligence

obotics 'Death of the office' homeworking claims exaggerated Anti-social robots harı increase social distanc

Artificial intelligence + Add to myFT

AI discovers antibiotics to treat drug-resistant diseases

Machine learning uncovers potent new drug able to kill 35 powerful bacteria

[29] Jonathan M.Stokes, et al. "A Deep Learning Approach to Antibiotic Discovery"

So far, we have talked about..

1. Graph Neural Network
2. Research questions in GNN architectures
3. GNN training strategy
4. Applications in science

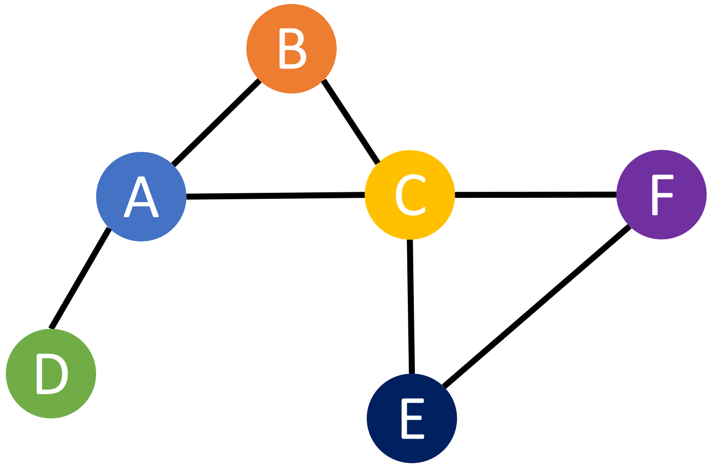
So far, we have talked about..

1. Graph Neural Network
2. Research questions in GNN architectures
3. GNN training strategy
4. Applications in science
5. **Recent Directions inspired by other disciplines**
   - **How to apply Transformers to graphs**
   - **How to utilize Large Language Models (LLMs) for graph tasks**

Advanced

Graph Transformer

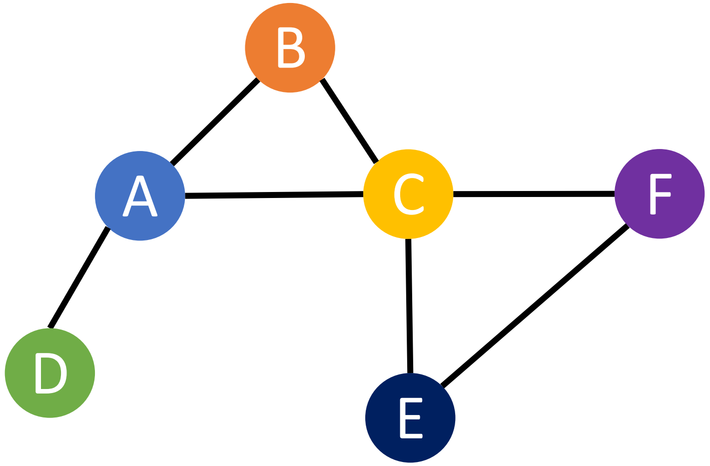
- Limitations of GNN
- Hard-coded structural inductive bias
  - Limited expressiveness (WL-test)
  - Oversmoothing
  - Oversquashing



Graph Transformer

- Aggregate from all nodes in a graph regardless of their connectivity

… Are we going to throw away the graph information?



Graph Transformer

- Graphormer[27]

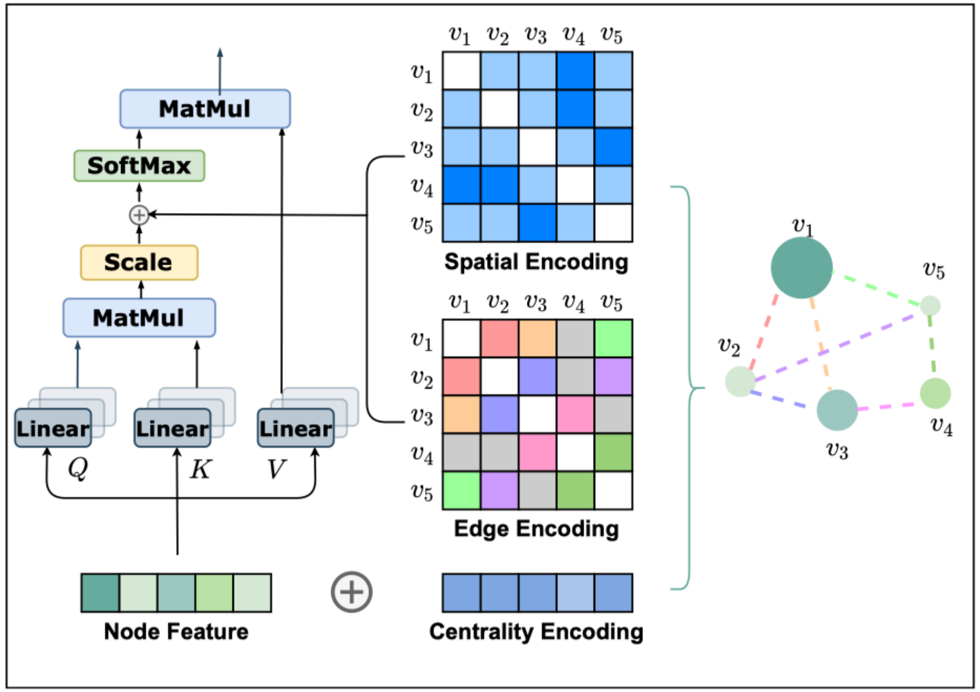
Degree information

$$h_i^{(0)} = x_i + \boxed{z^-_{\deg^-(v_i)} + z^+_{\deg^+(v_i)}}$$

$$A_{ij} = \frac{(h_i W_Q)(h_j W_K)^T}{\sqrt{d}} + \boxed{b_{\phi(v_i,v_j)} + c_{ij},} \text{ where } c_{ij} = \frac{1}{N}\sum_{n=1}^{N} x_{e_n}(w_n^E)^T$$

Spatial distance information

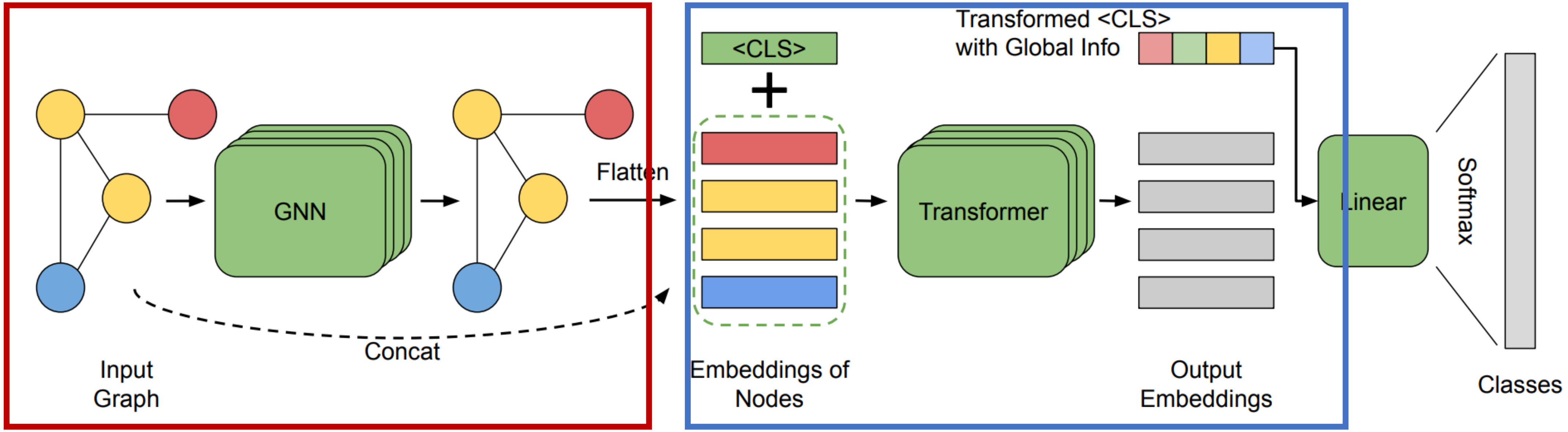
Edge attribute information



[27] Chengxuan Ying, et al. "Do Transformers Really Perform Bad for Graph Representation?"

Graph Transformer

- GraphTrans[28]



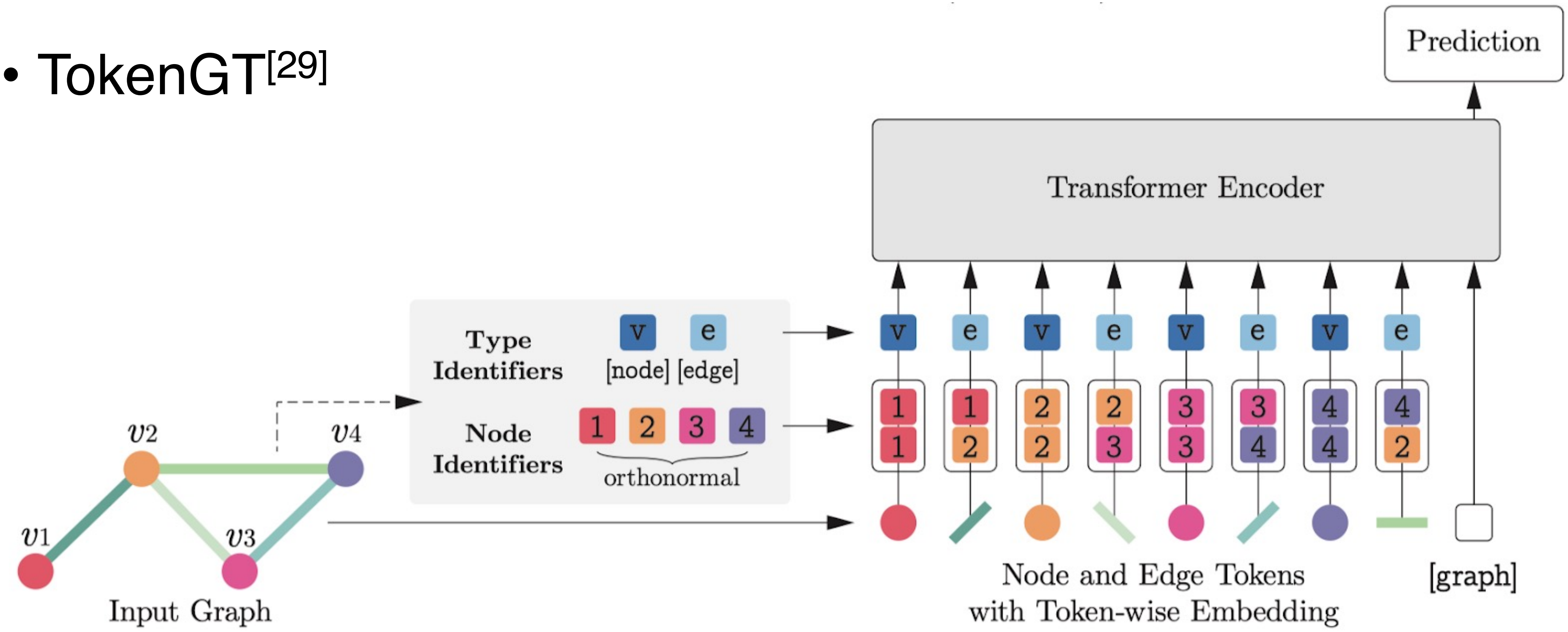
Local information

Global information

[28] Zhanghao Wu, et al. "Representing Long-Range Context for Graph Neural Networks with Global Attention"

Graph Transformer

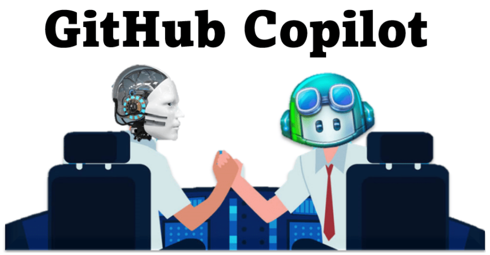
- TokenGT[29]

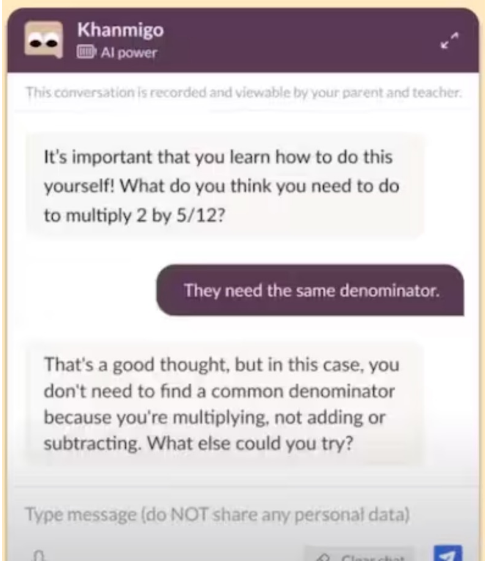


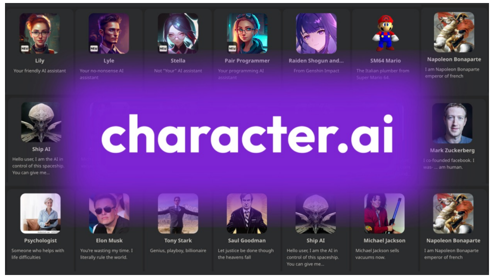
[29] Jinwoo Kim, et al. "Pure Transformers are Powerful Graph Learners"

LLM + GNN

- Large Language Model (LLM)
- ”Inflection point in AI”, “Dawn of Artificial General Intelligence”







Rowan Cheung @rowancheung · Follow

I just watched GPT-4 turn a hand-drawn sketch into a functional website.

This is insane.

Ammaar Reshi @ammaar · Follow

Can GPT-4 code an entire game for you? Yes, yes it can.

Pietro Schirano @skirano · Follow

I don't care that it's not AGI, GPT-4 is an incredible and transformative technology.

I recreated the game of Pong in under 60 seconds.
It was my first try.

Dan Shipper @danshipper · Follow

GPT-4 does drug discovery.

Give it a currently available drug and it can:

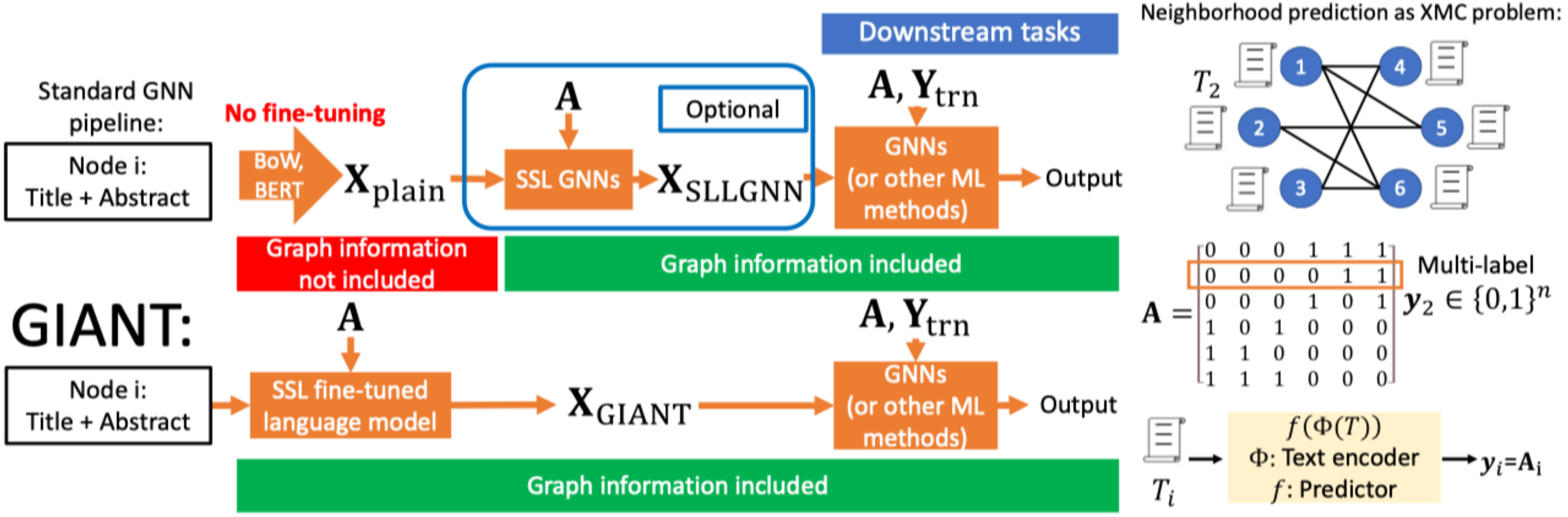
LLM + GNN

- Large Language Model (LLM)
- ”Inflection point in AI”, “Dawn of Artificial General Intelligence”

Can we improve GNN
using LLMs?

LLM + GNN

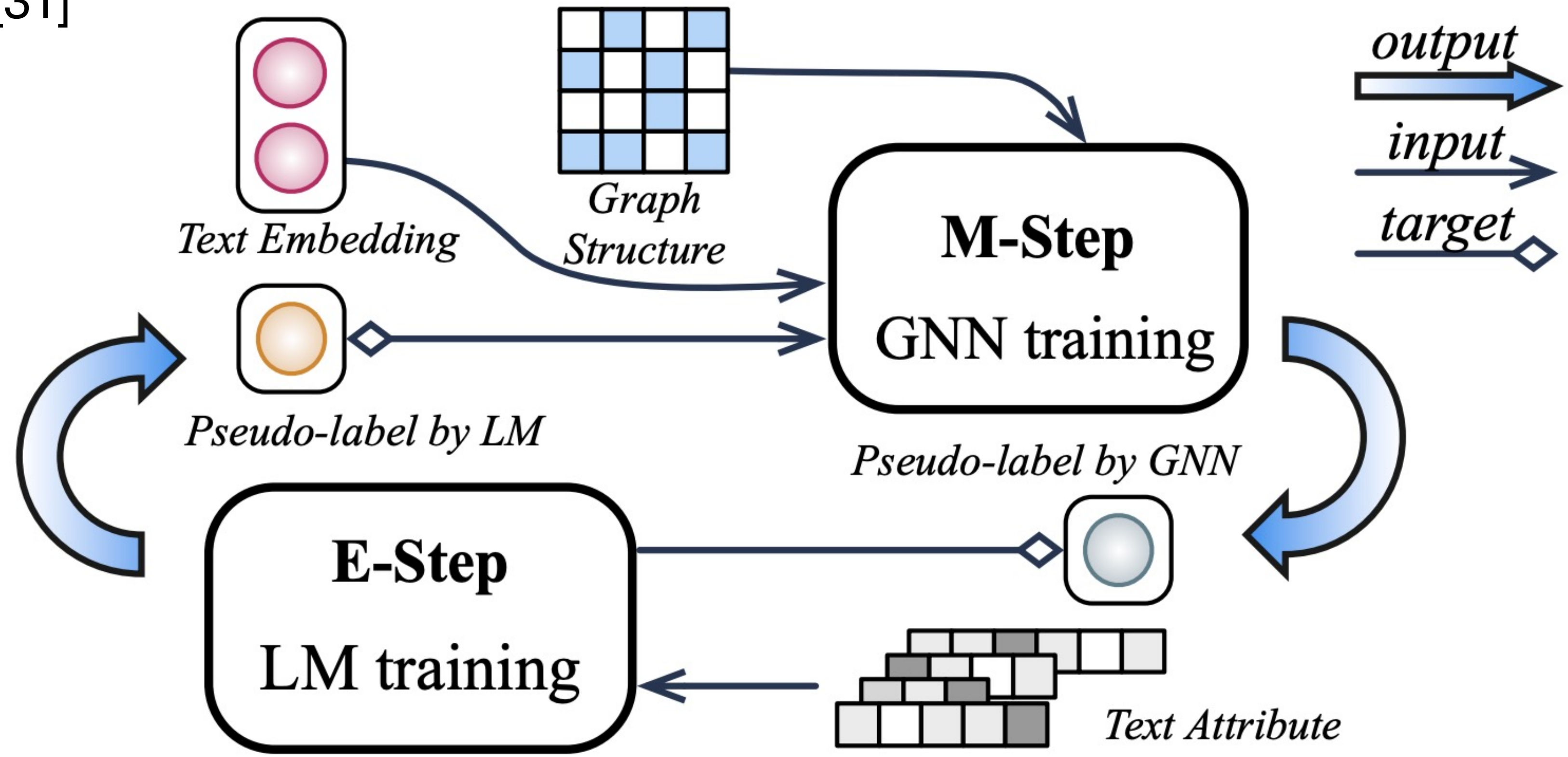
- GIANT[30]



[30] Eli Chien, et al. "Node Feature Extraction by Self-Supervised Multi-scale Neighborhood Prediction"

LLM + GNN

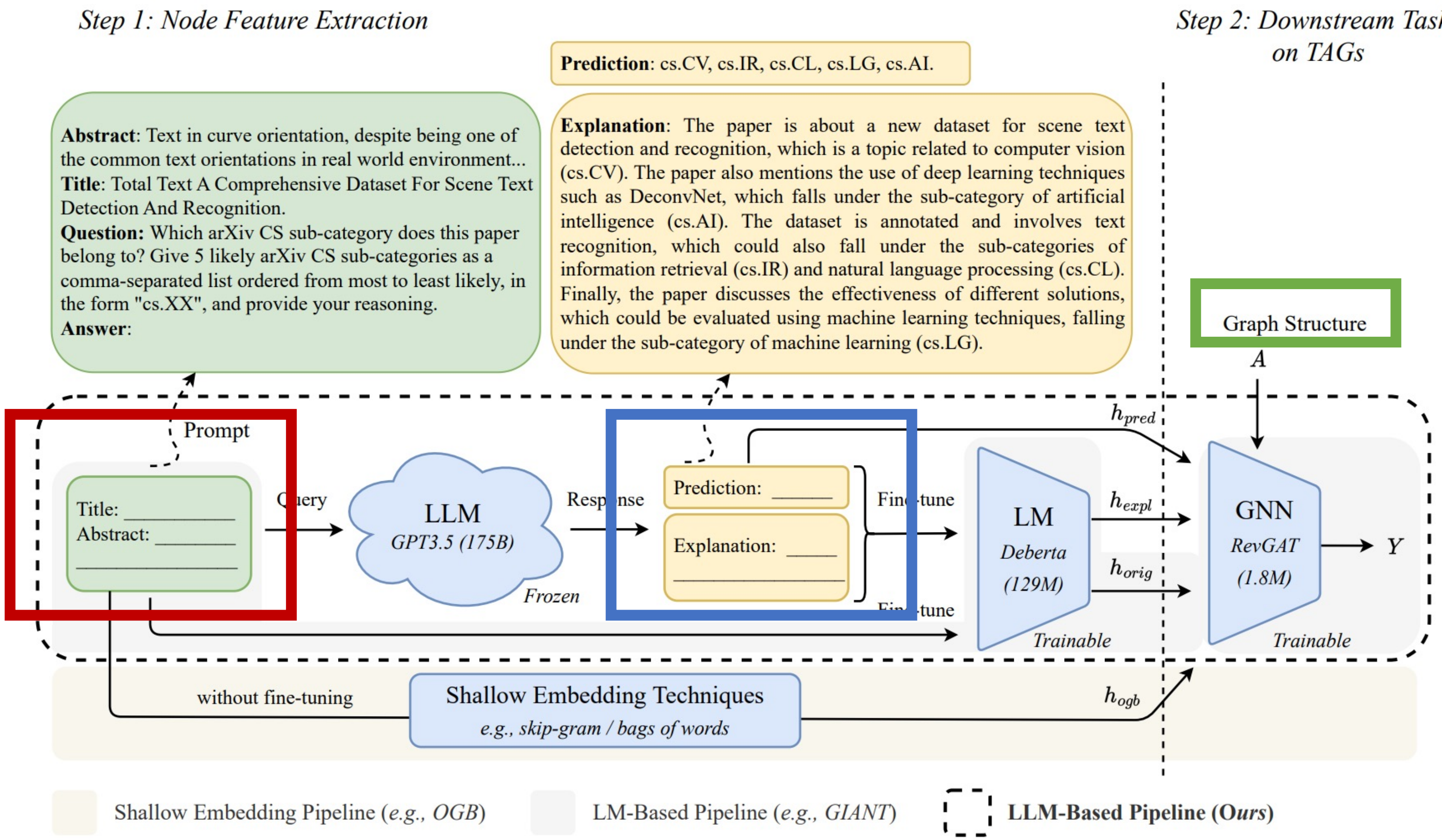
- GLEM[31]



[31] Jianan Zhao, et al. "Learning on Large-scale Text-attributed Graphs via Variational Inference"

LLM + GNN

- TAPE[32]



[32] Xiaoxin He, et al. "Harnessing Explanations: LLM-to-LM Interpreter for Enhanced Text-Attributed Graph Representation Learning"

Still many open problems..

- And many more chances to do groundbreaking research
- Diverse types of graphs
  - 3-dimensional graphs
  - Temporal graphs
  - Multimodal graphs
- Diverse types of architecture
  - Graph Convolution Networks
  - Graph Transformers
  - LLMs

Thank you!

Questions?

minjiy@cs.cmu.edu | https://www.minjiyoon.xyz