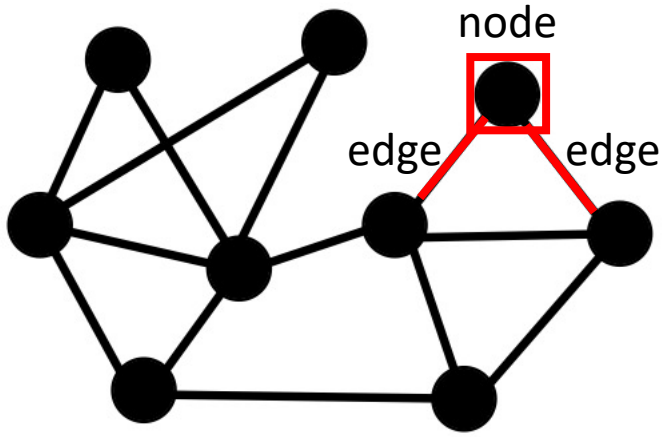
Graph Representation

Neural Networks Design And Application

Correlation between data



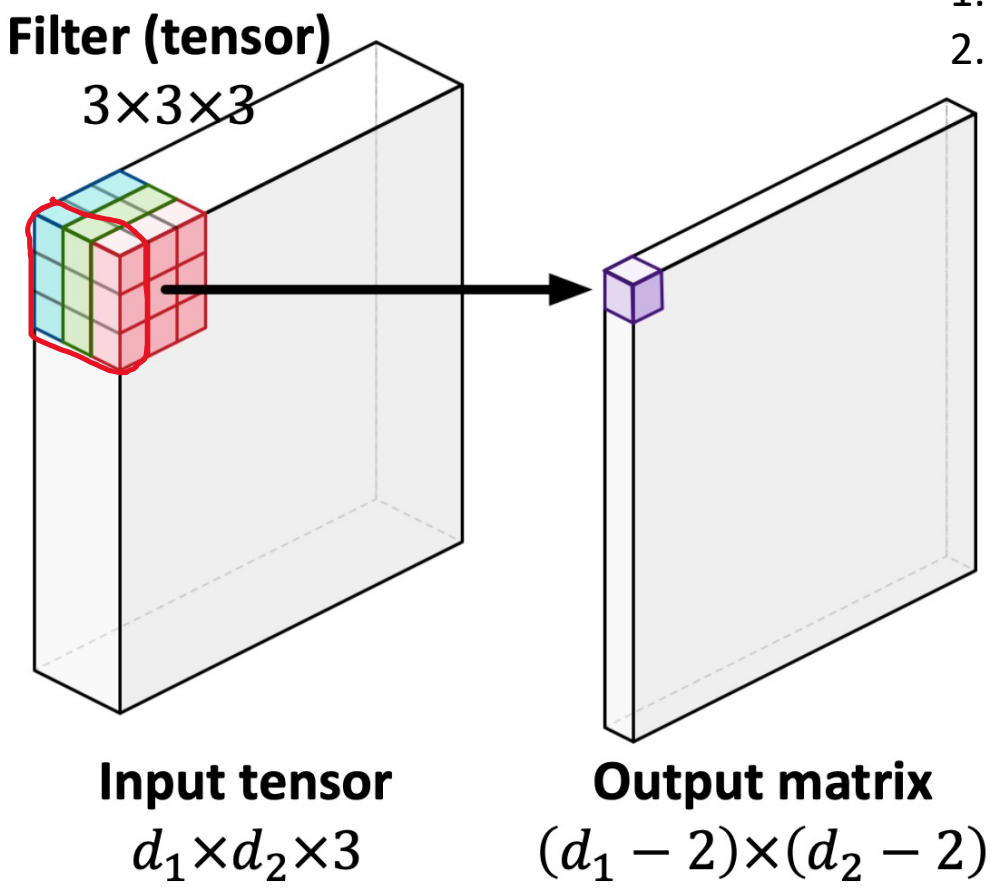
undirected graph

Image credit http://web.stanford.edu/class/cs224w/slides/01-intro.pdf

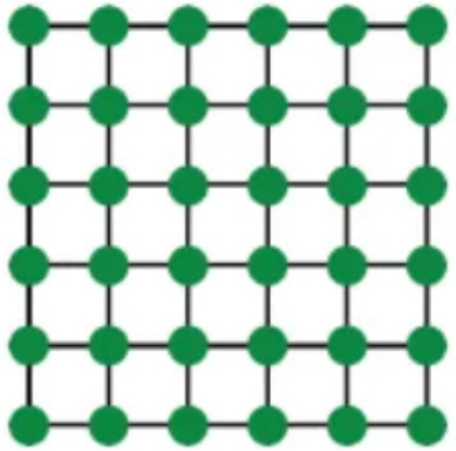
Correlation between data

Q: is there correlation between:

1. locations?
2. channels?



Regard feature maps as a grid



Pixel values or feature values

Correlation between data

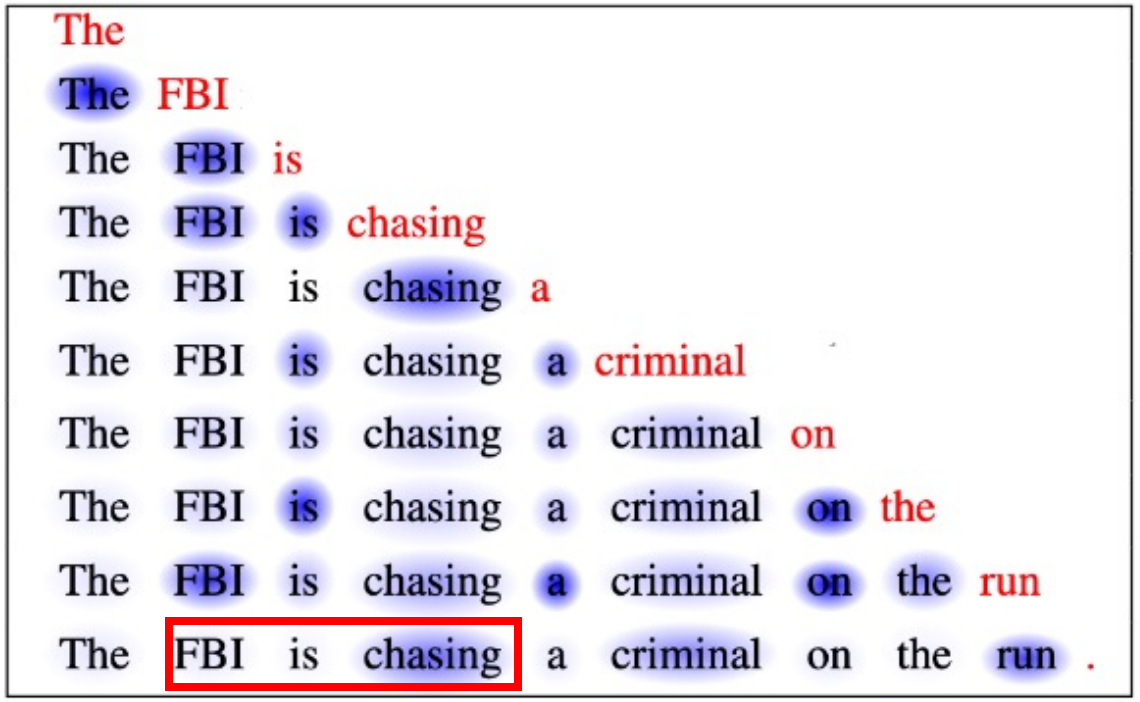


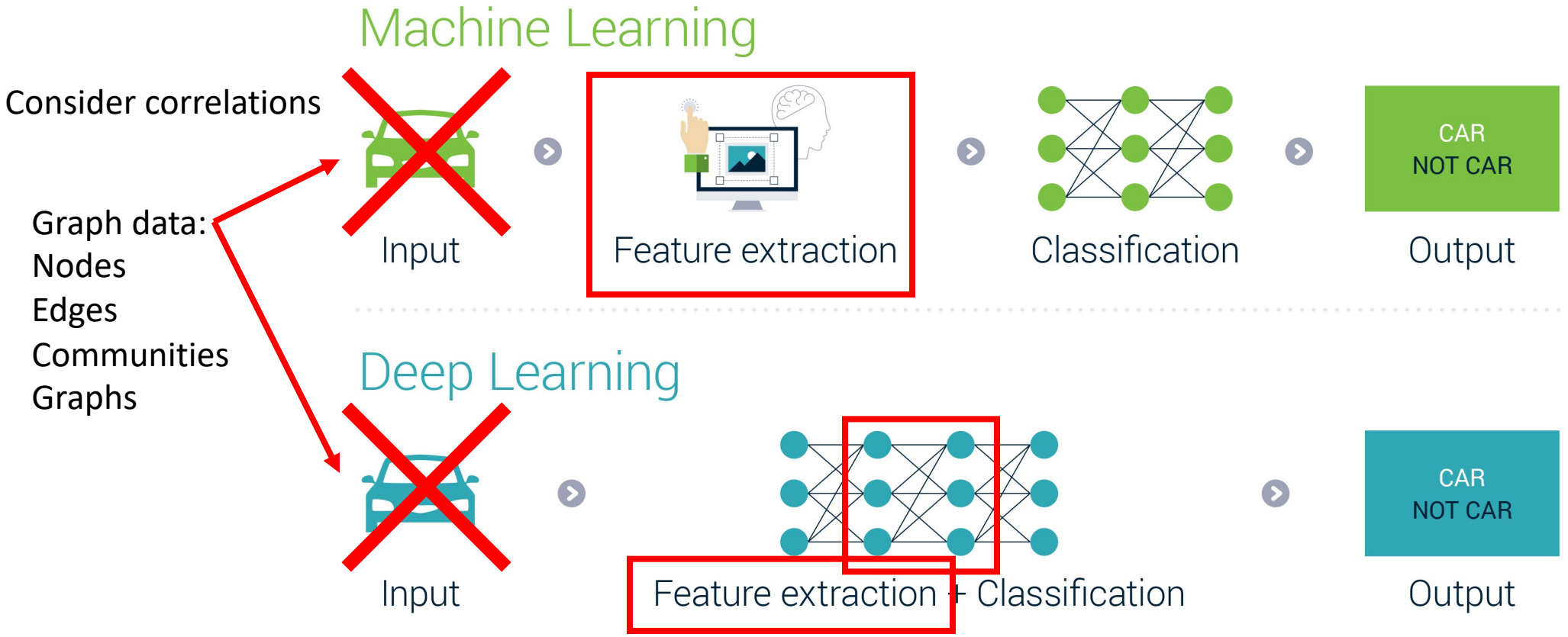
Figure is from the paper " Long Short-Term Memory-Networks for Machine Reading."

The FBI is → FBI is chasings

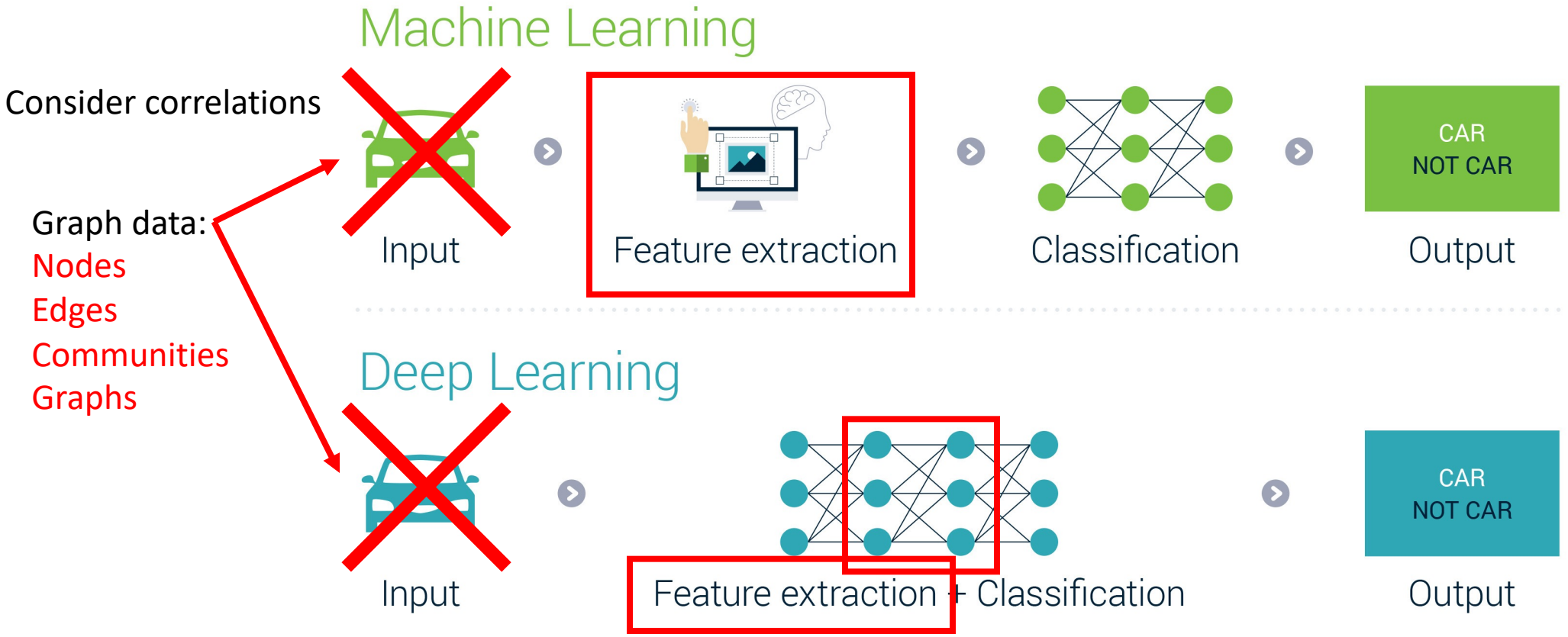
Sequence data 

Q: what if we need more complicated correlation?

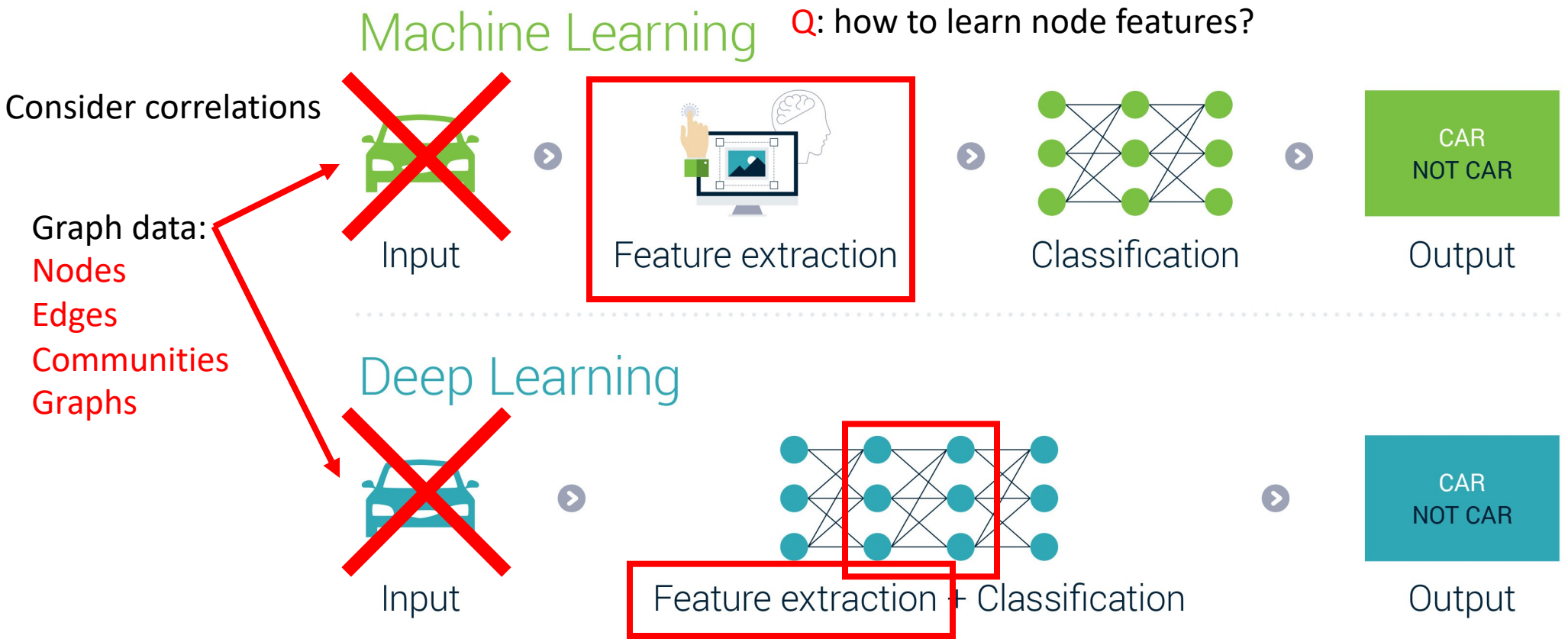
Machine learning tasks for graph data



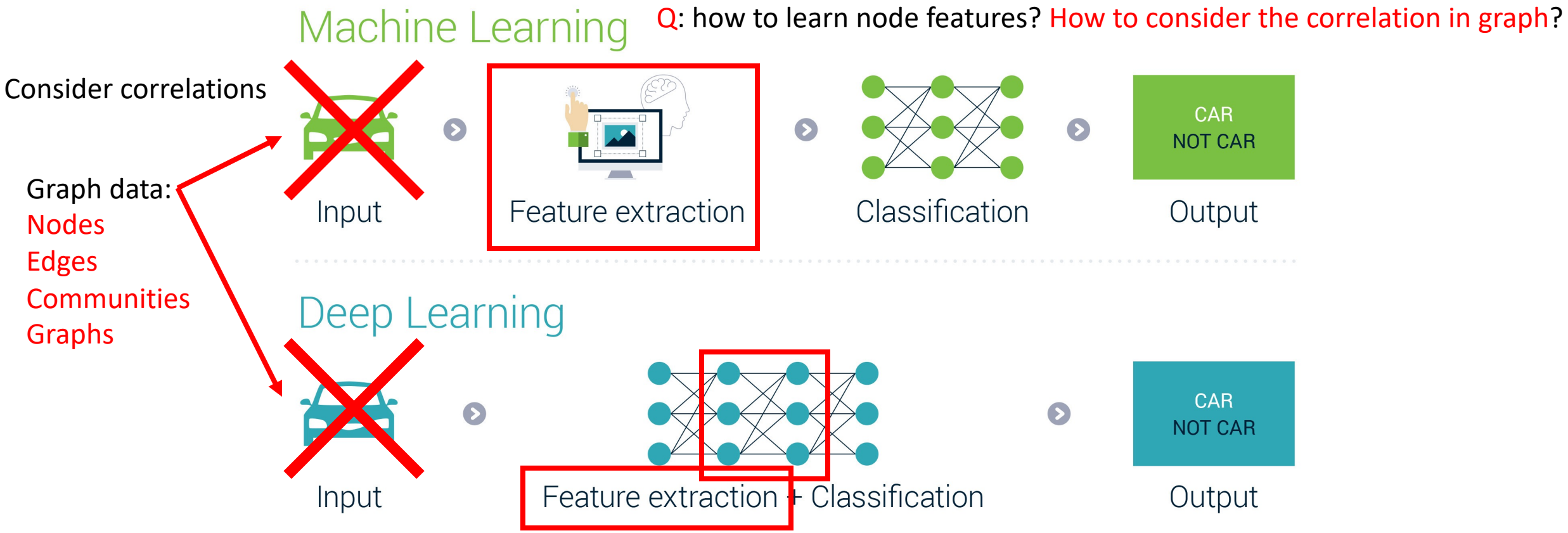
Machine learning tasks for graph data



Machine learning tasks for graph data



Machine learning tasks for graph data



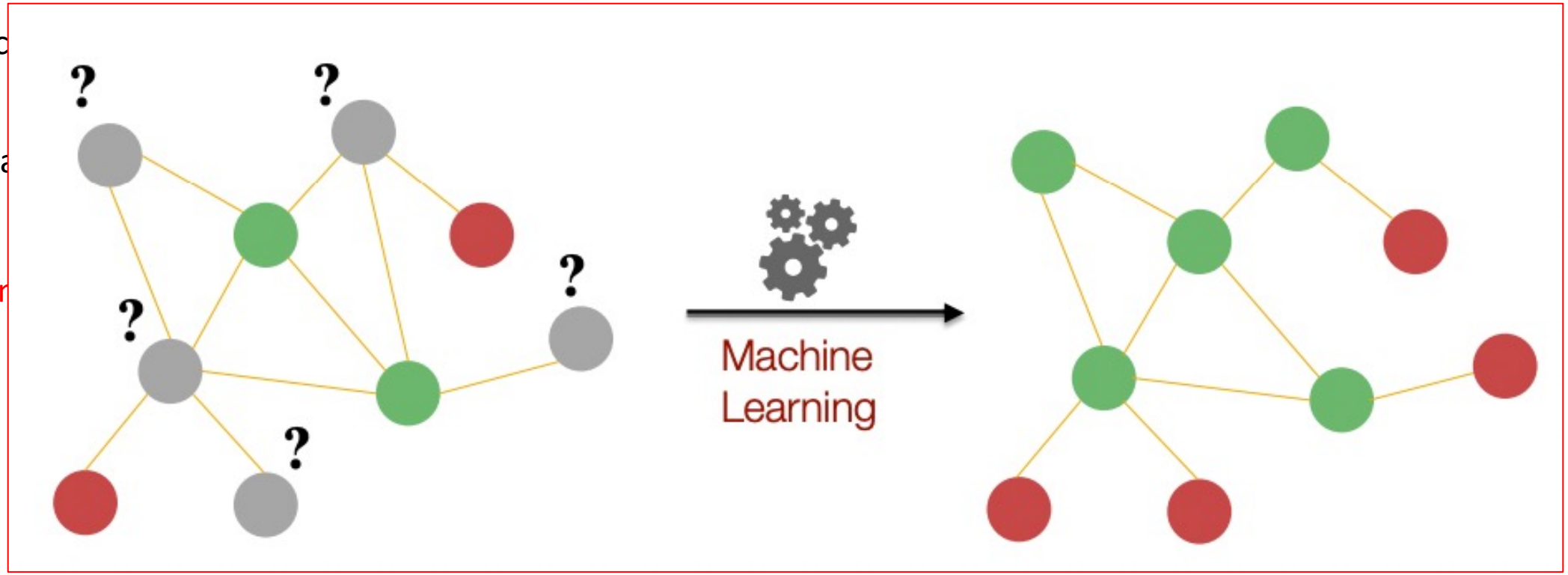
Machine learning tasks for graph data

Machine Learning

Q: how to learn node features? How to consider the correlation in graph?

Consider c

Graph da
Nodes
Edges
Commur
Graphs



Machine learning tasks for graph data

Machine Learning

Q: how to learn node features? How to consider the correlation in graph?

Consider c

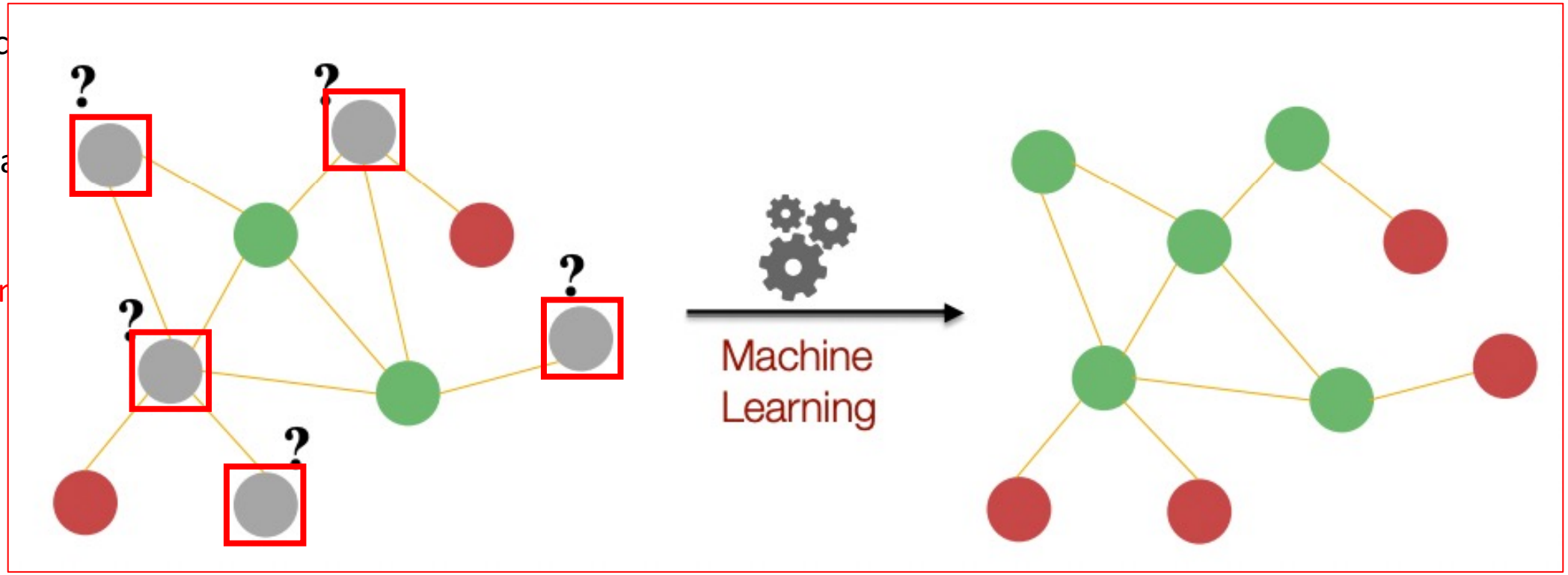
Graph da

Nodes

Edges

Commur

Graphs



Machine learning tasks for graph data

Machine Learning

Q: how to learn node features? How to consider the correlation in graph?

Consider c

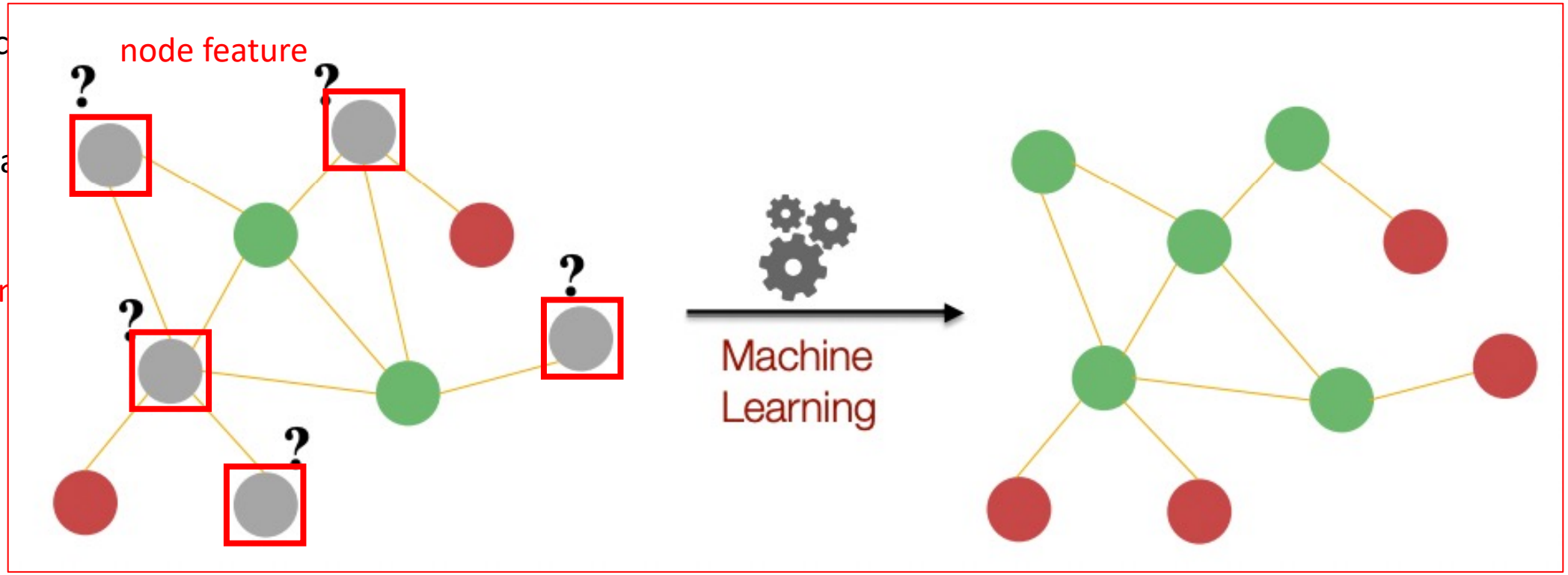
Graph da

Nodes

Edges

Commur

Graphs



Machine learning tasks for graph data

Machine Learning

Q: how to learn node features? How to consider the correlation in graph?

Consider c

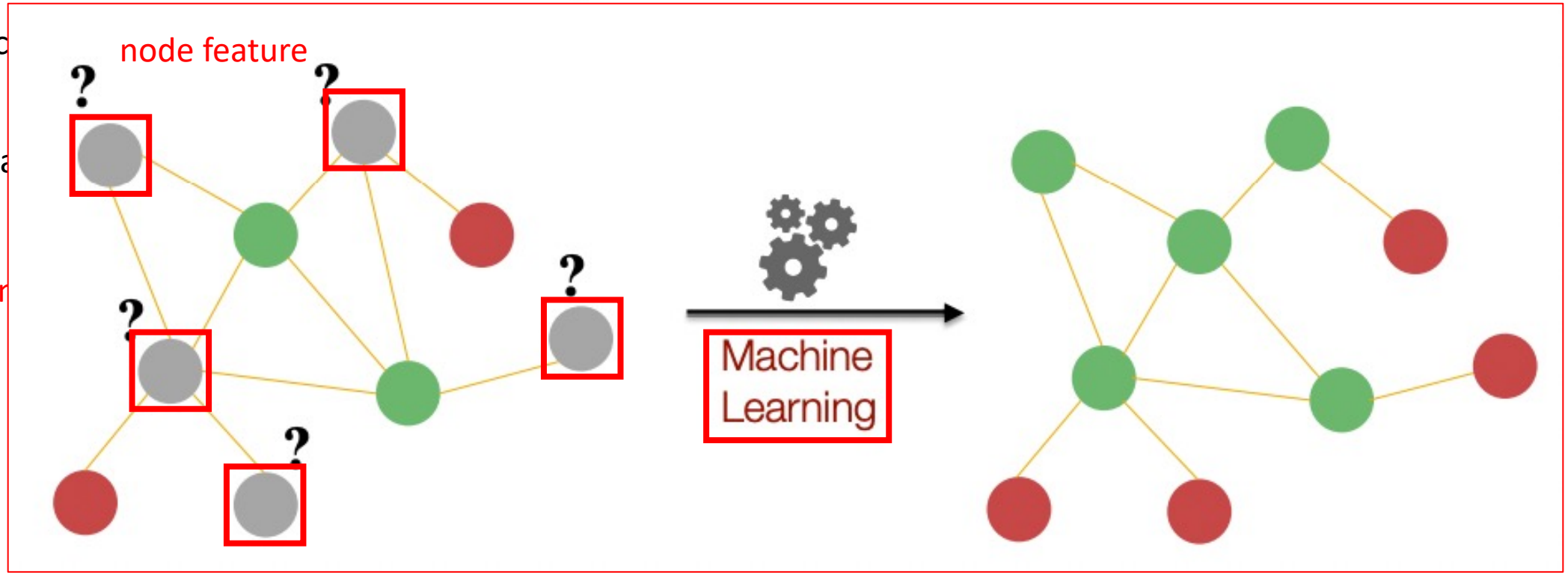
Graph da

Nodes

Edges

Commur

Graphs



Machine learning tasks for graph data

Machine Learning

Q: how to learn node features? How to consider the correlation in graph?

Consider c

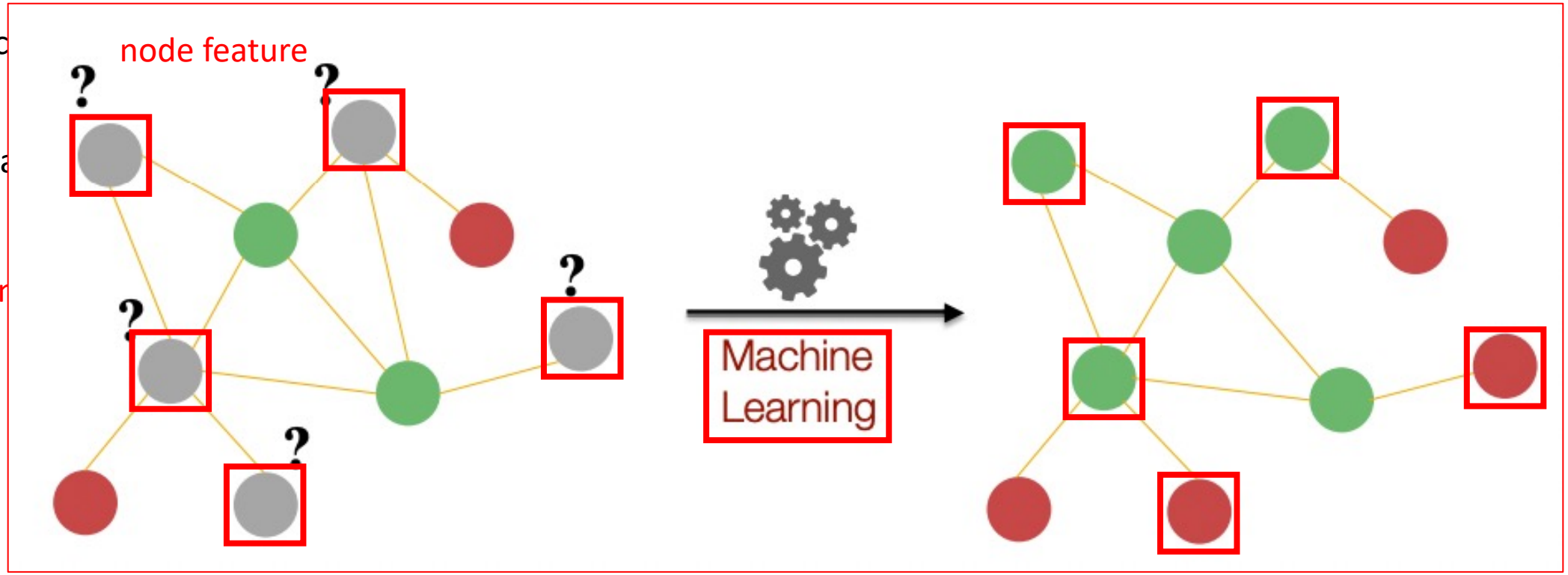
Graph da

Nodes

Edges

Commur

Graphs



Node degree

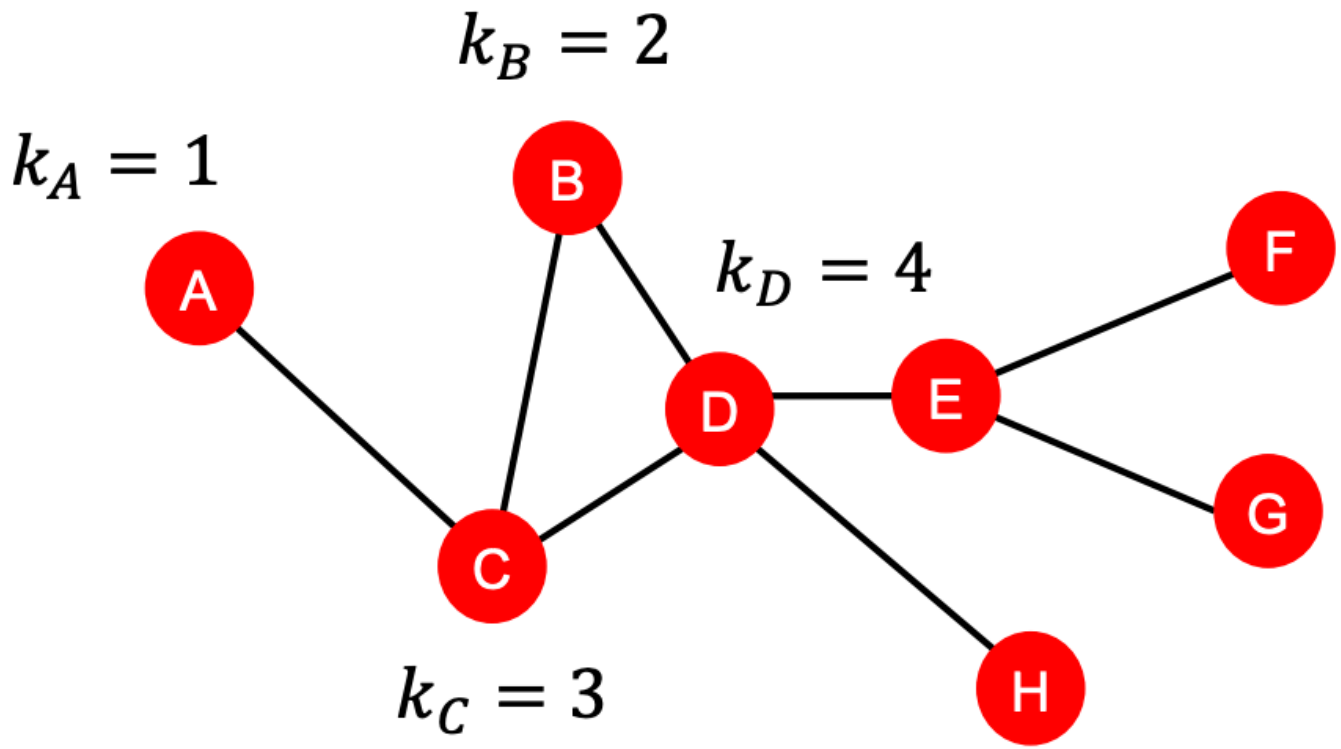


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Node degree

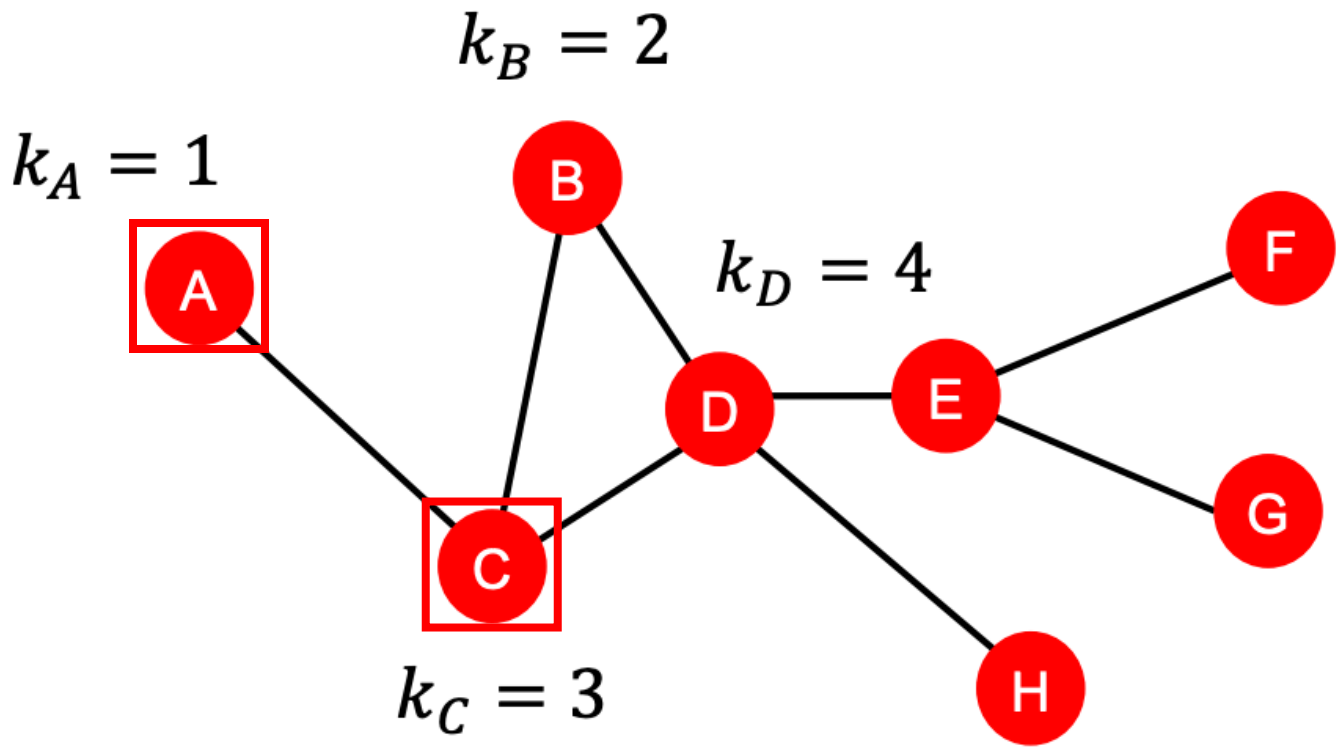


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Node degree

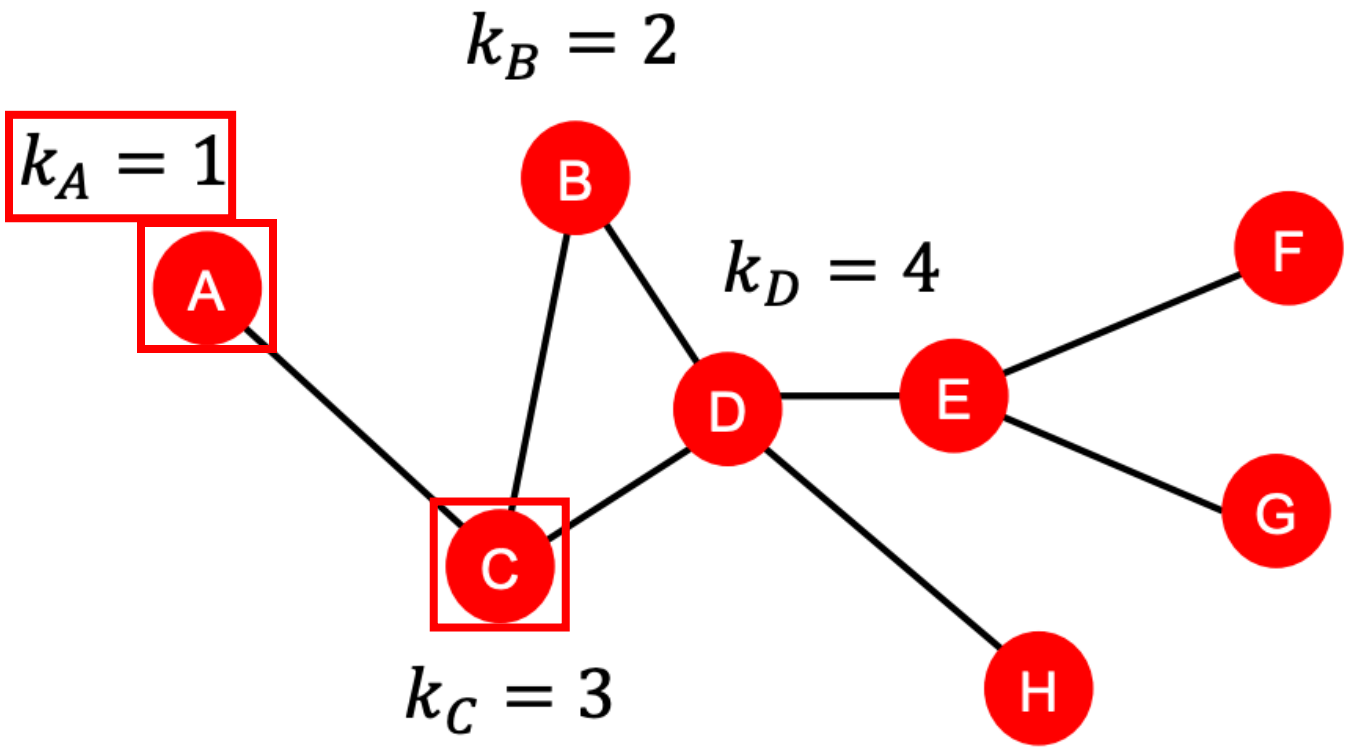


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Node degree

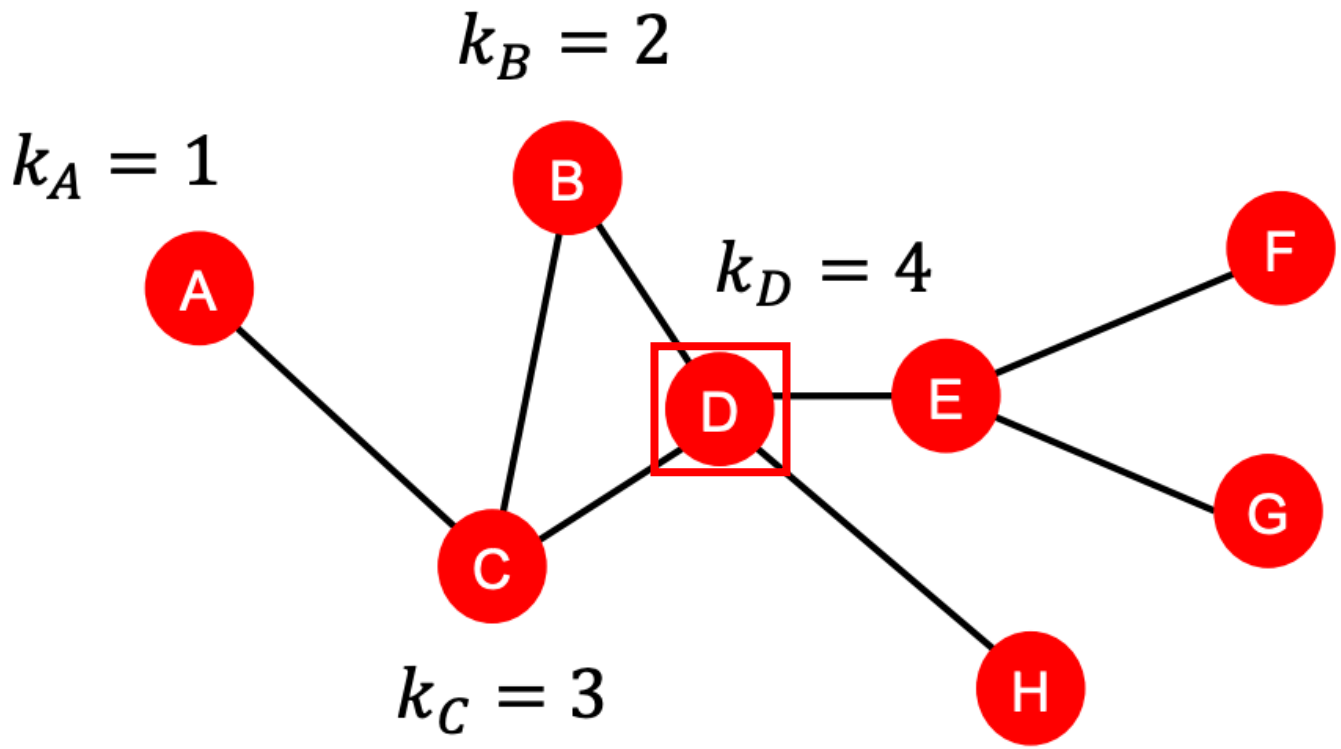


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Node degree

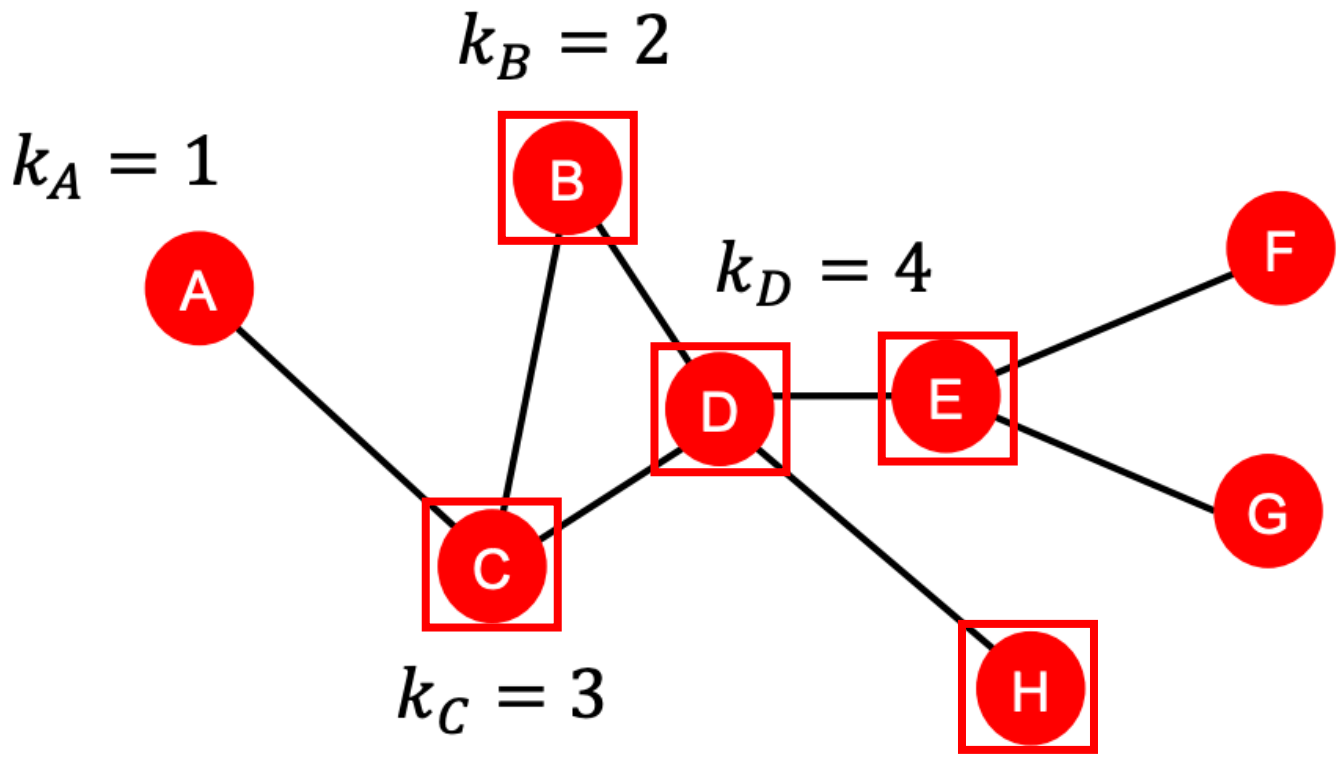


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Node degree

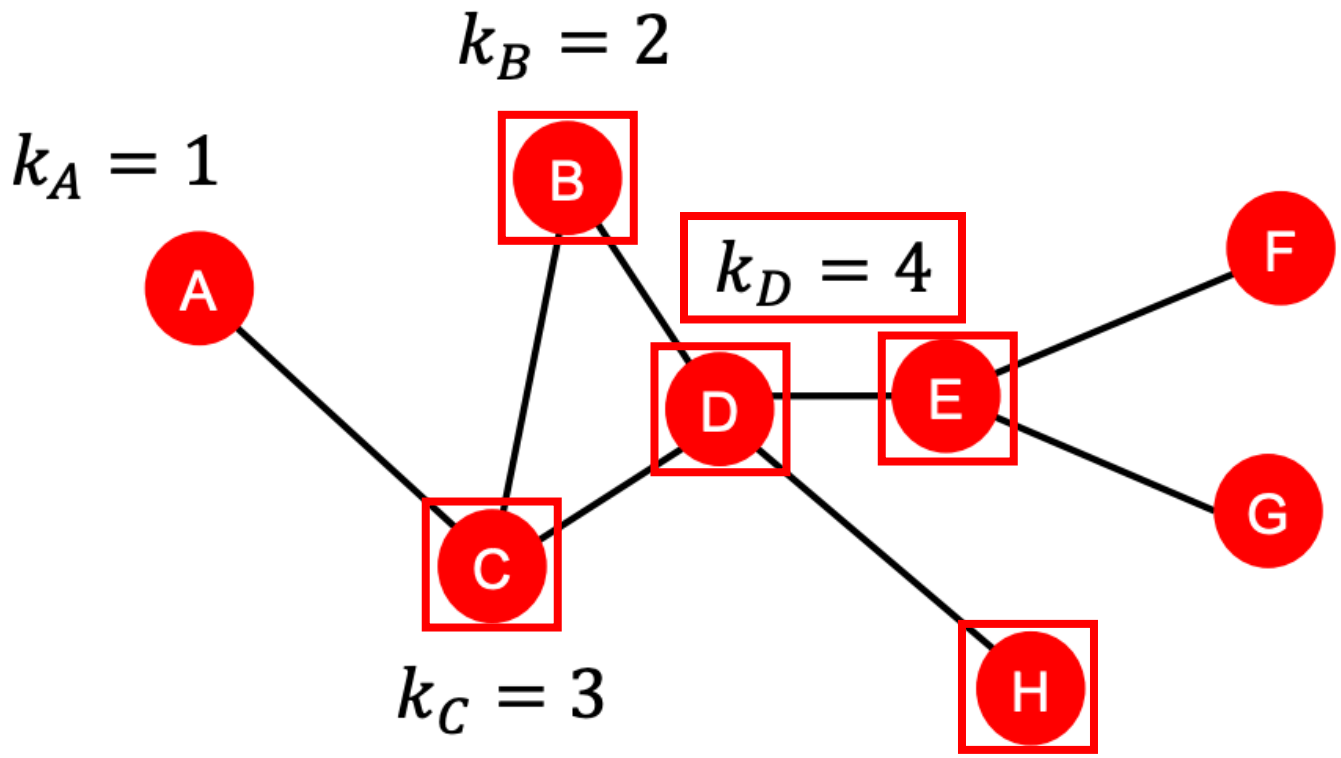
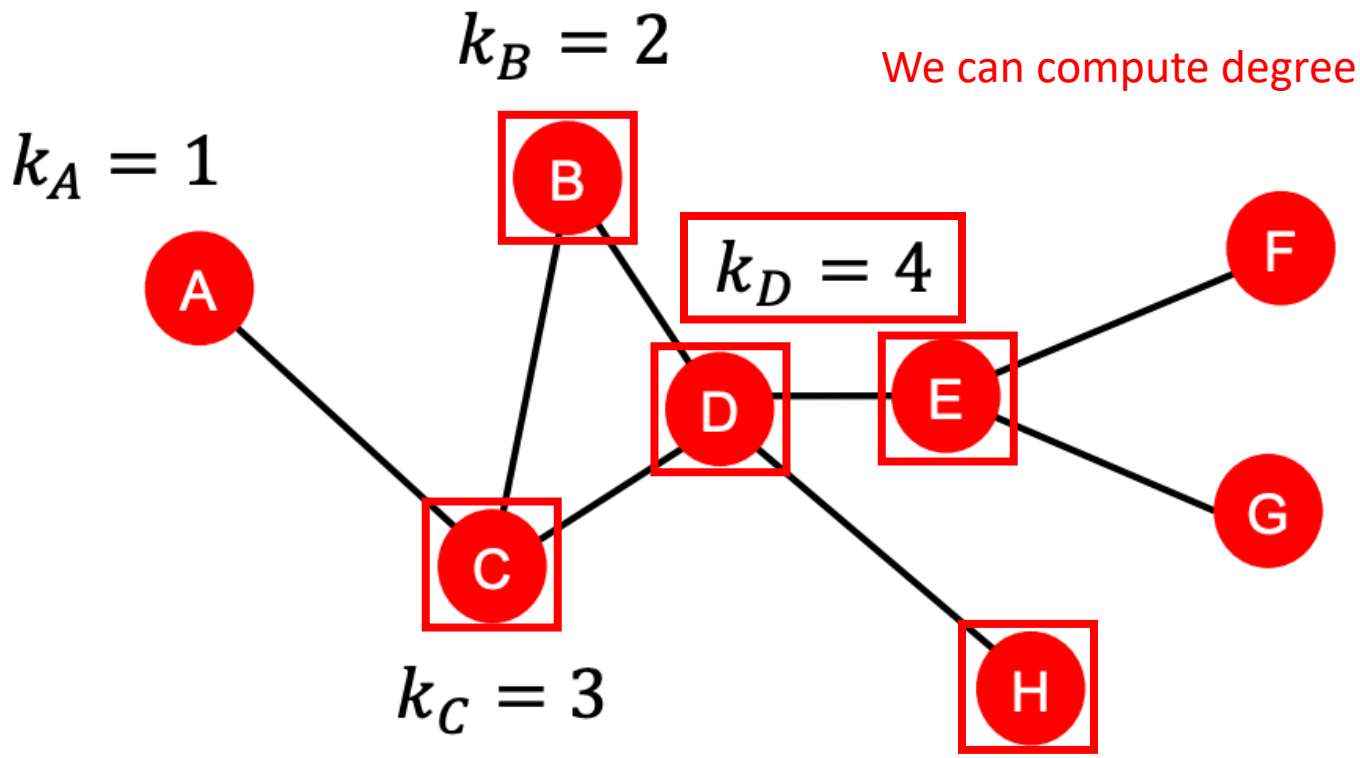


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

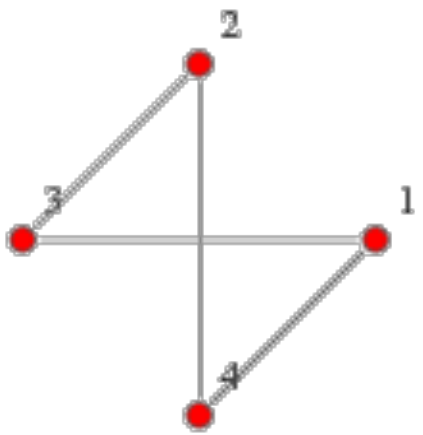
Node degree

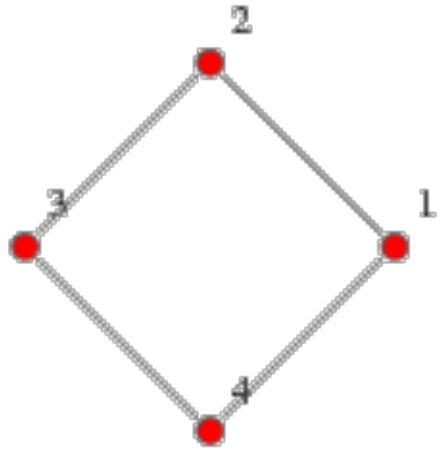


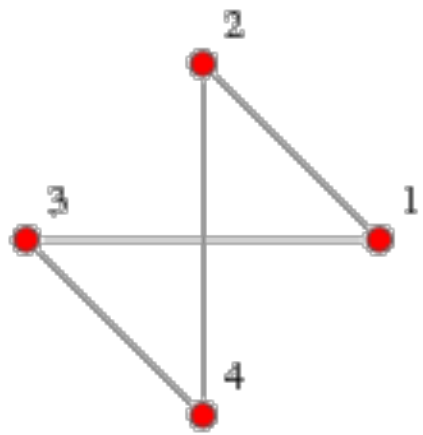
We can compute degree for each node

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Adjacency matrix

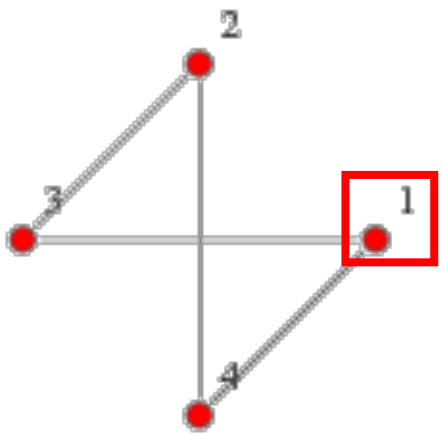


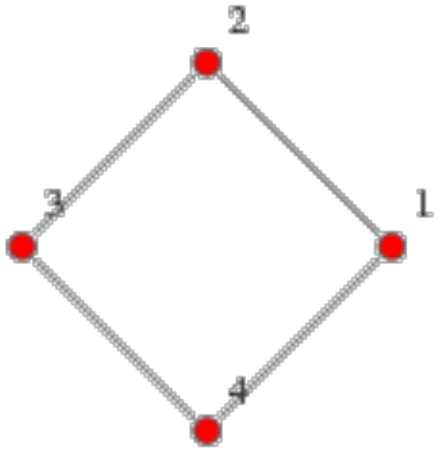


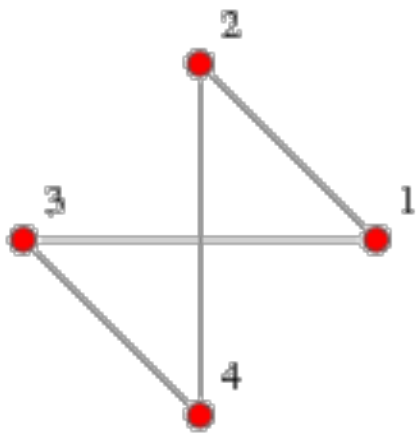


$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix

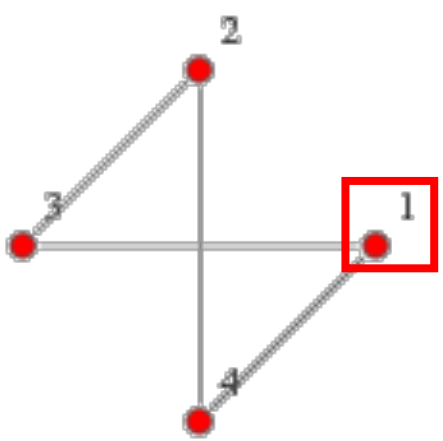


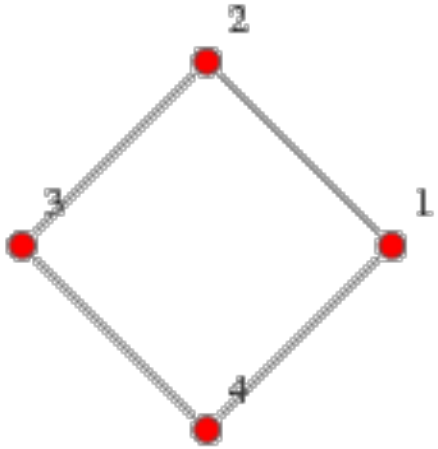


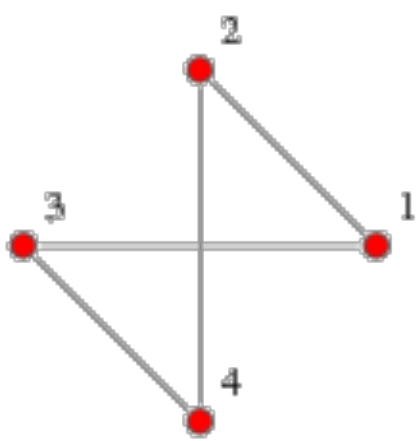


$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix



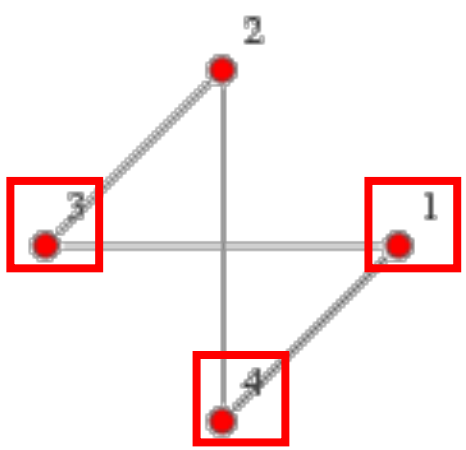


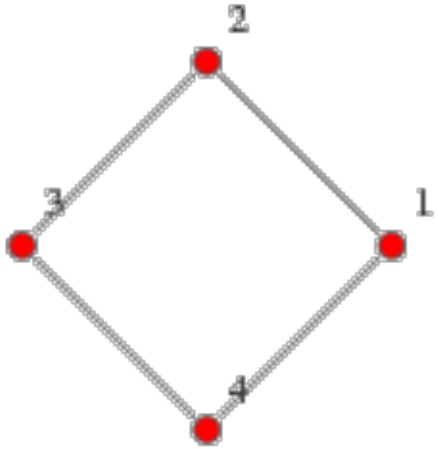


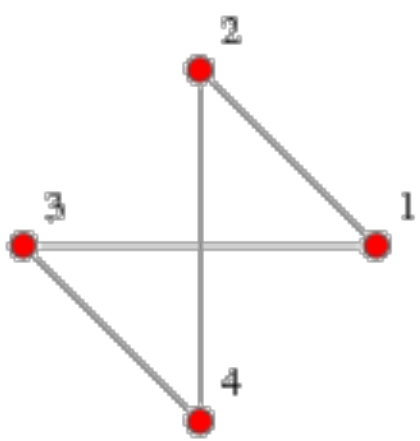
Q: which nodes are connected to node #1?

$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix



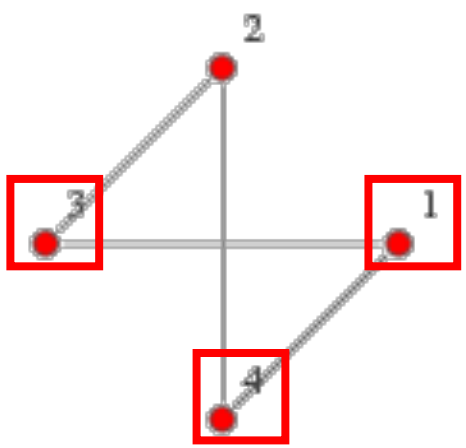


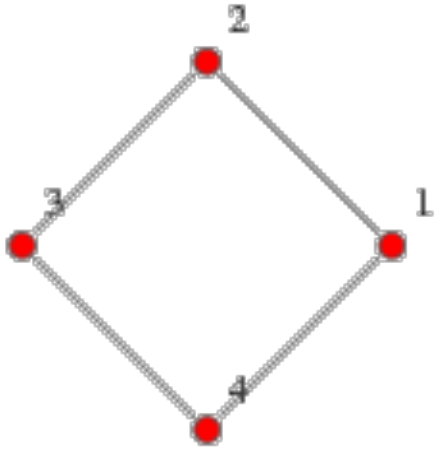


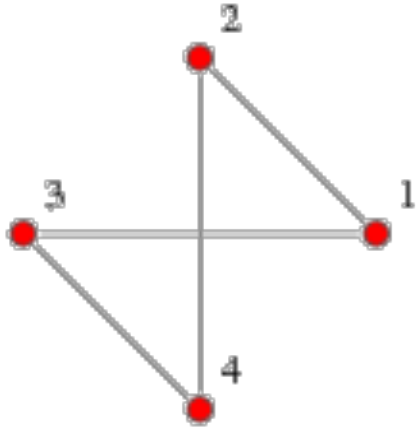
Q: which nodes are connected to node #1?

$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix



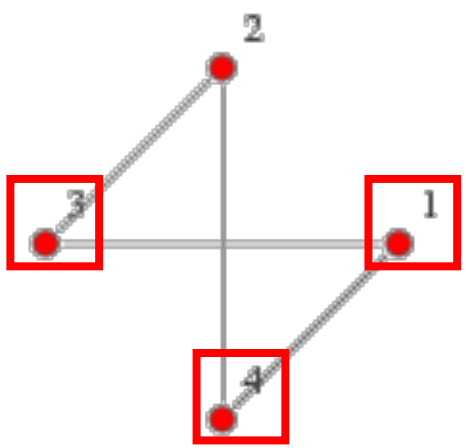


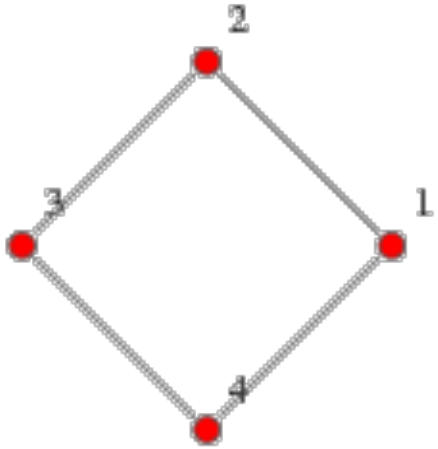


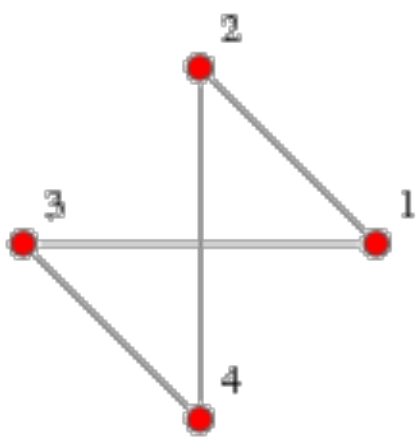
Q: which nodes are connected to node #1?

$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix



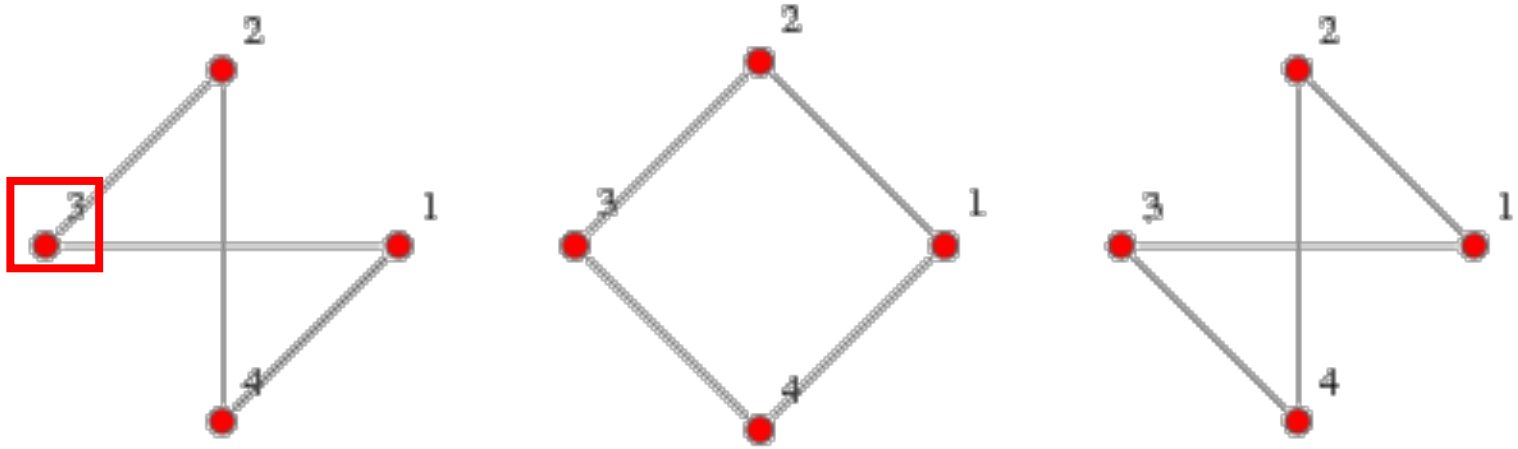




Q: which nodes are connected to node #1?

$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

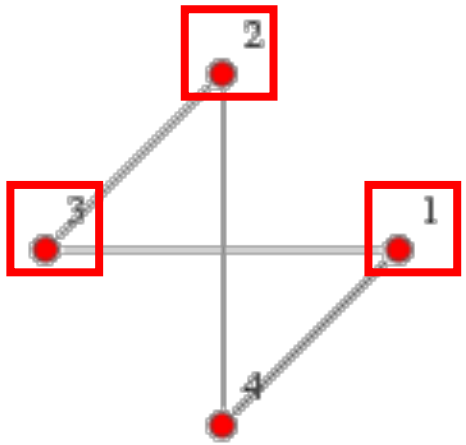
Adjacency matrix

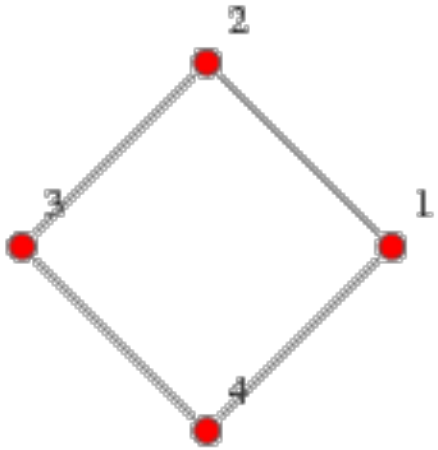


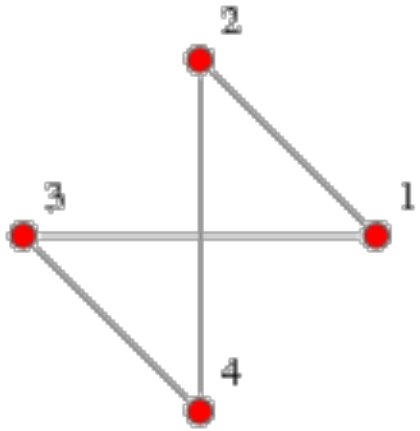
Q: which nodes are connected to node #3?

$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix



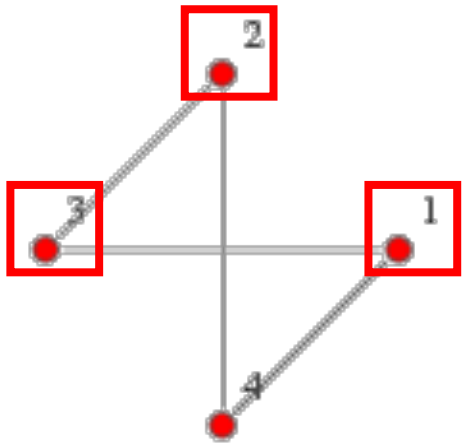


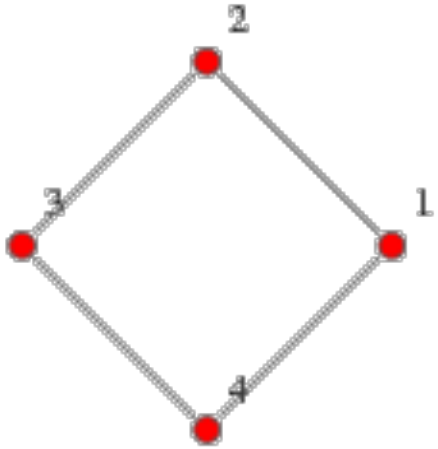


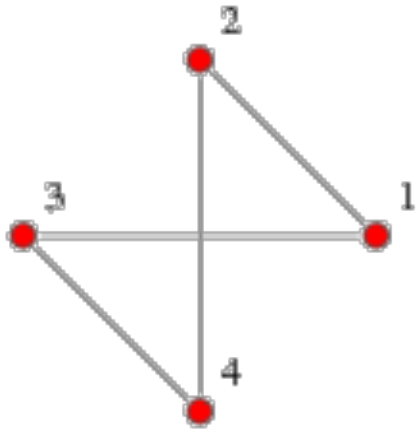
Q: which nodes are connected to node #3?

$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix



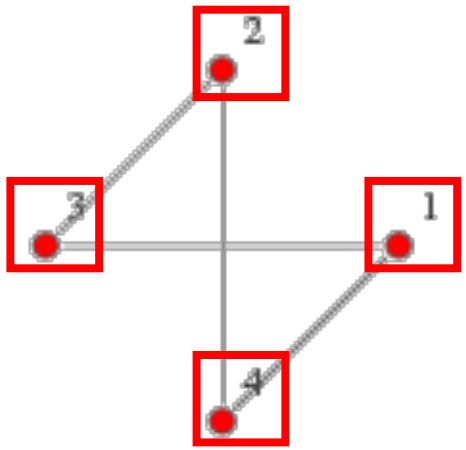


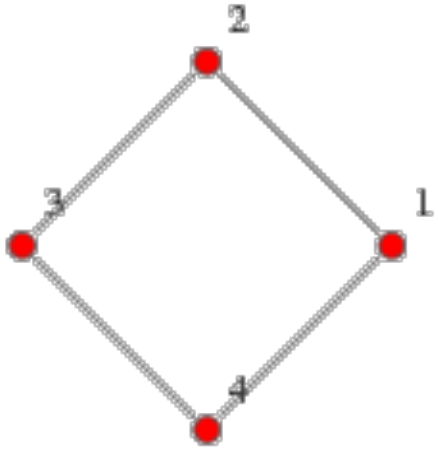


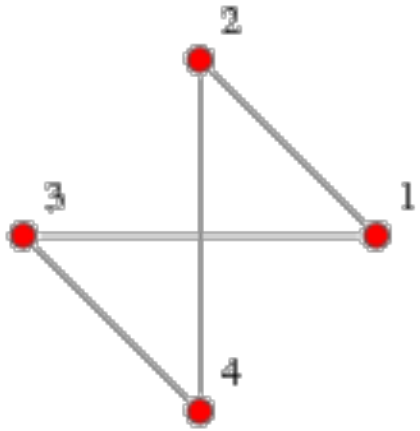
Q: which nodes are connected to node #3?

$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix





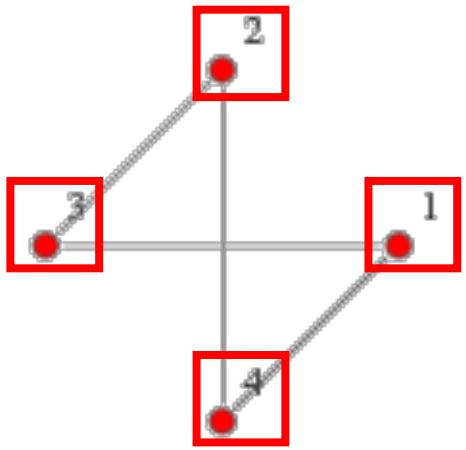


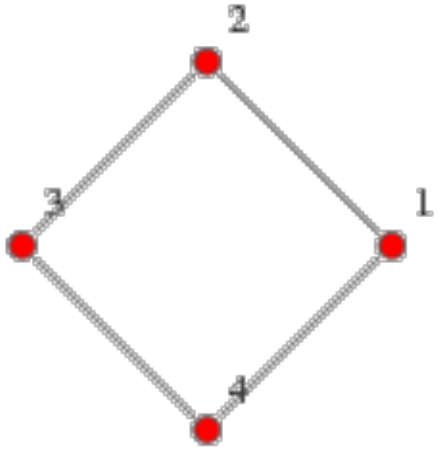
$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

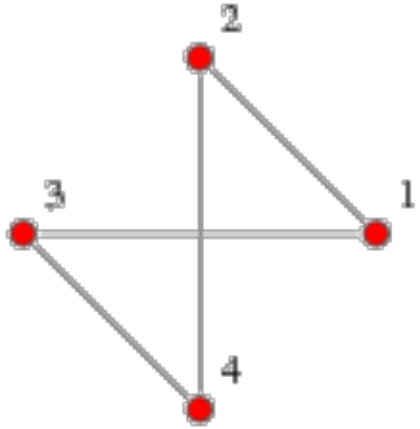
$$\begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix



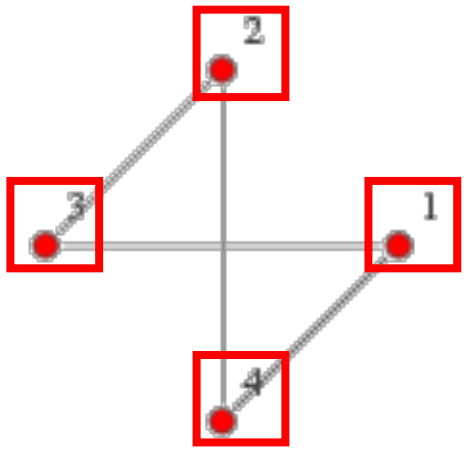


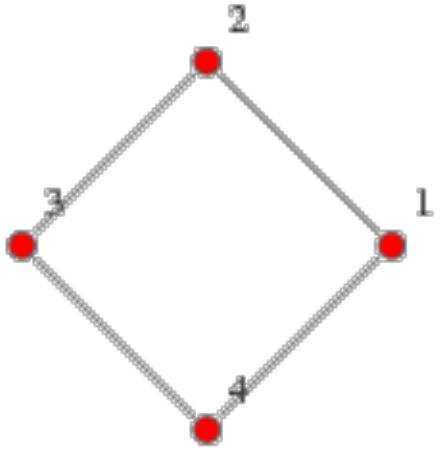


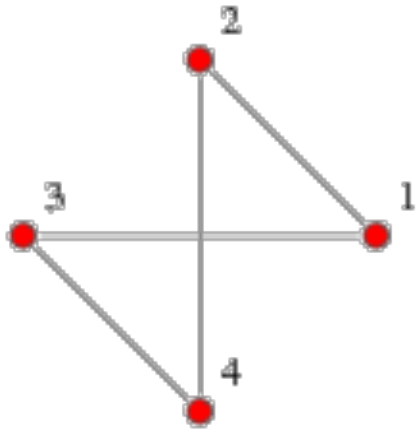
$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Q: What do entries on diagonal stand for?

Adjacency matrix





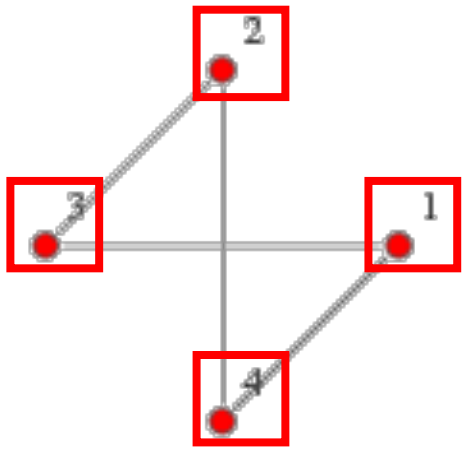


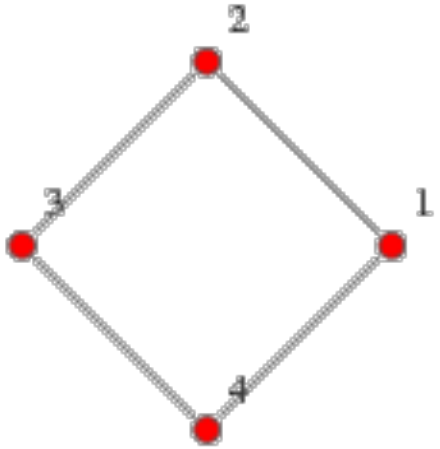
$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

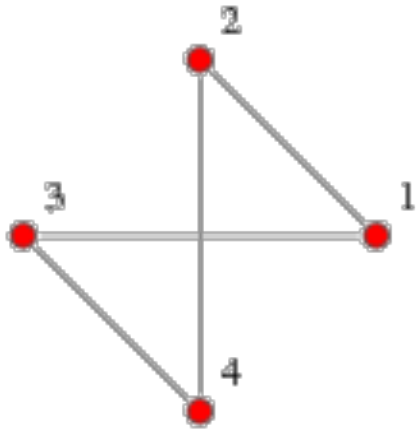
$$\begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix





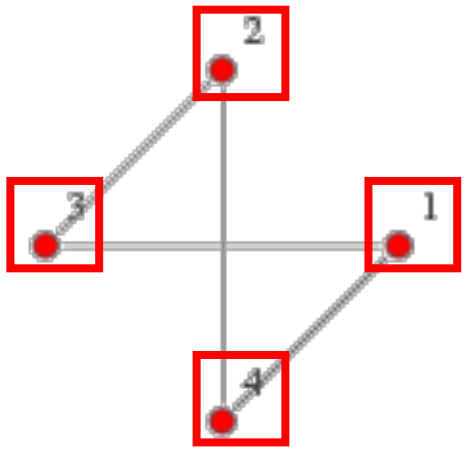


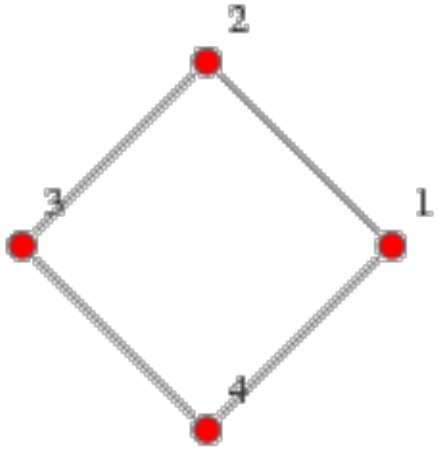
$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

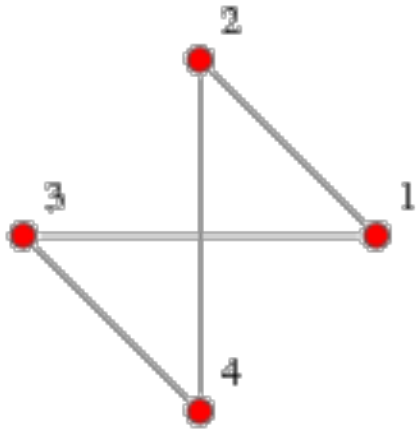
$$\begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Adjacency matrix



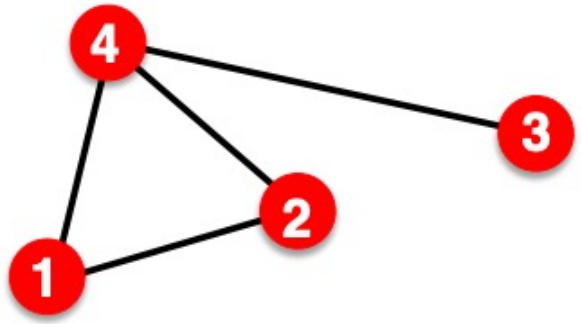




Symmetric

$$\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

Encoder-decoder for graph data

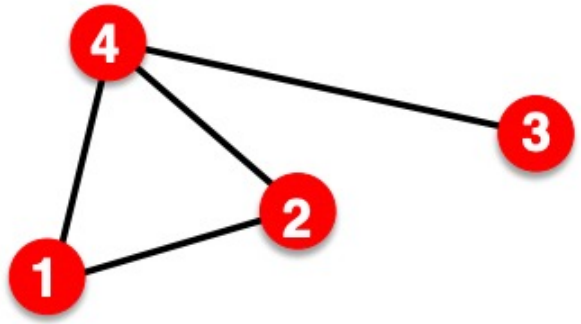


V: {1, 2, 3, 4}

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



V: {1, 2, 3, 4}

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

Q: can we learn node features with the correlation in the adjacency matrix?

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data

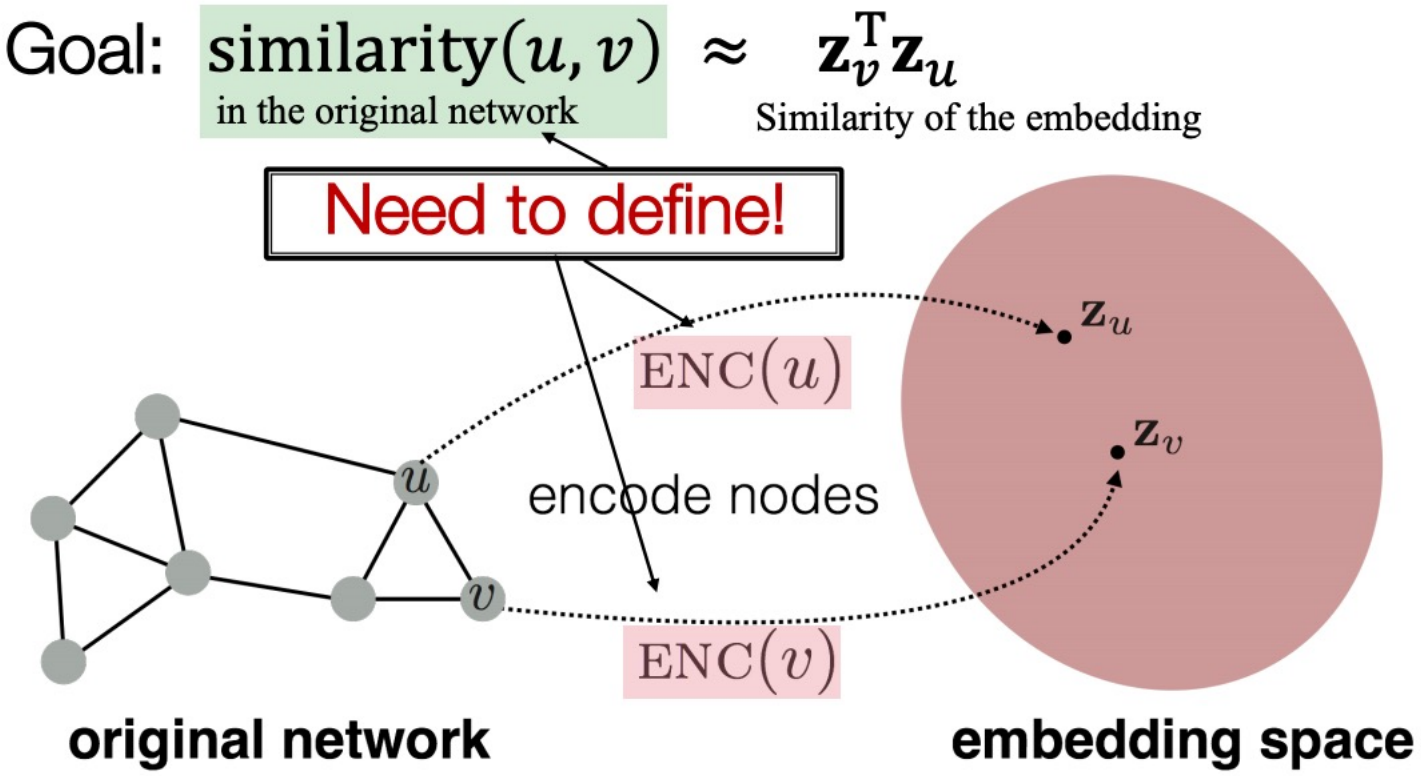


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data

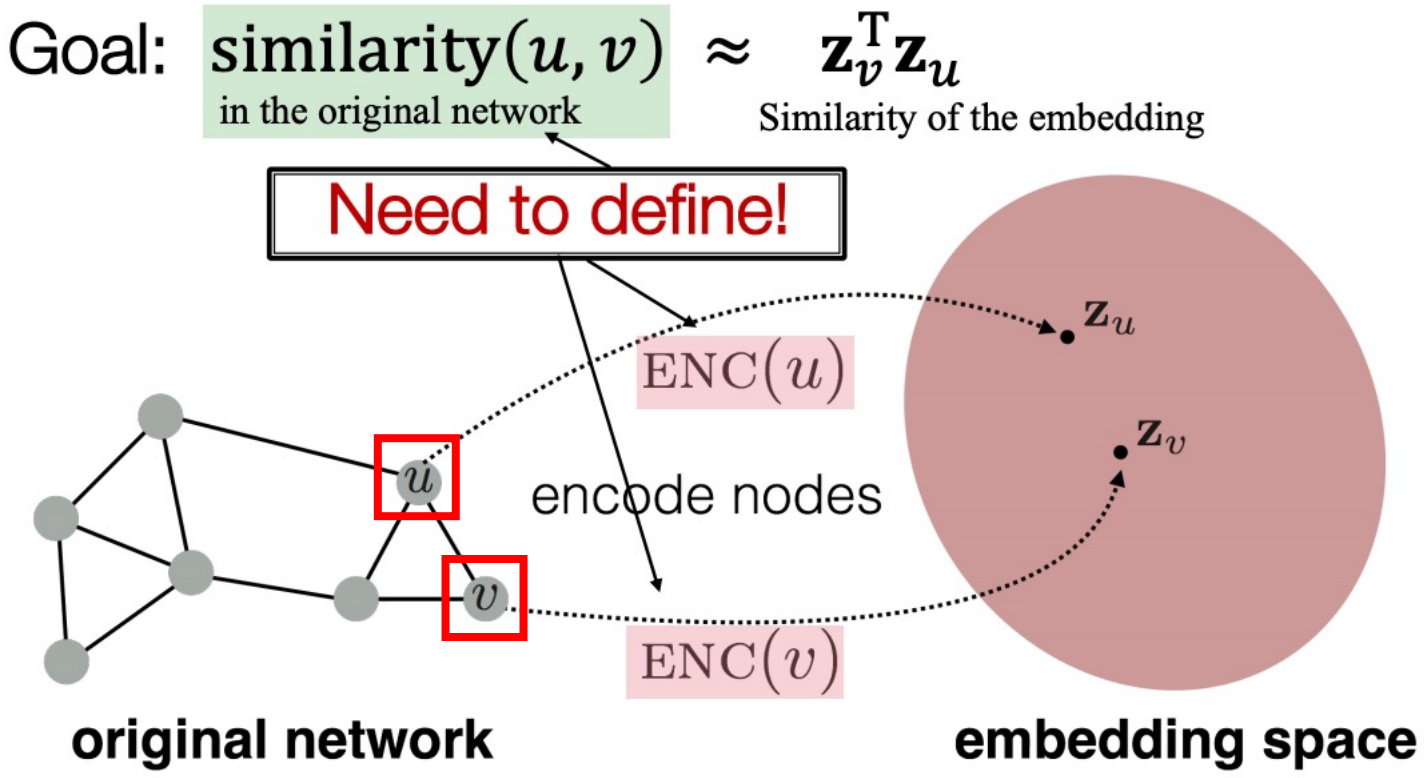


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data

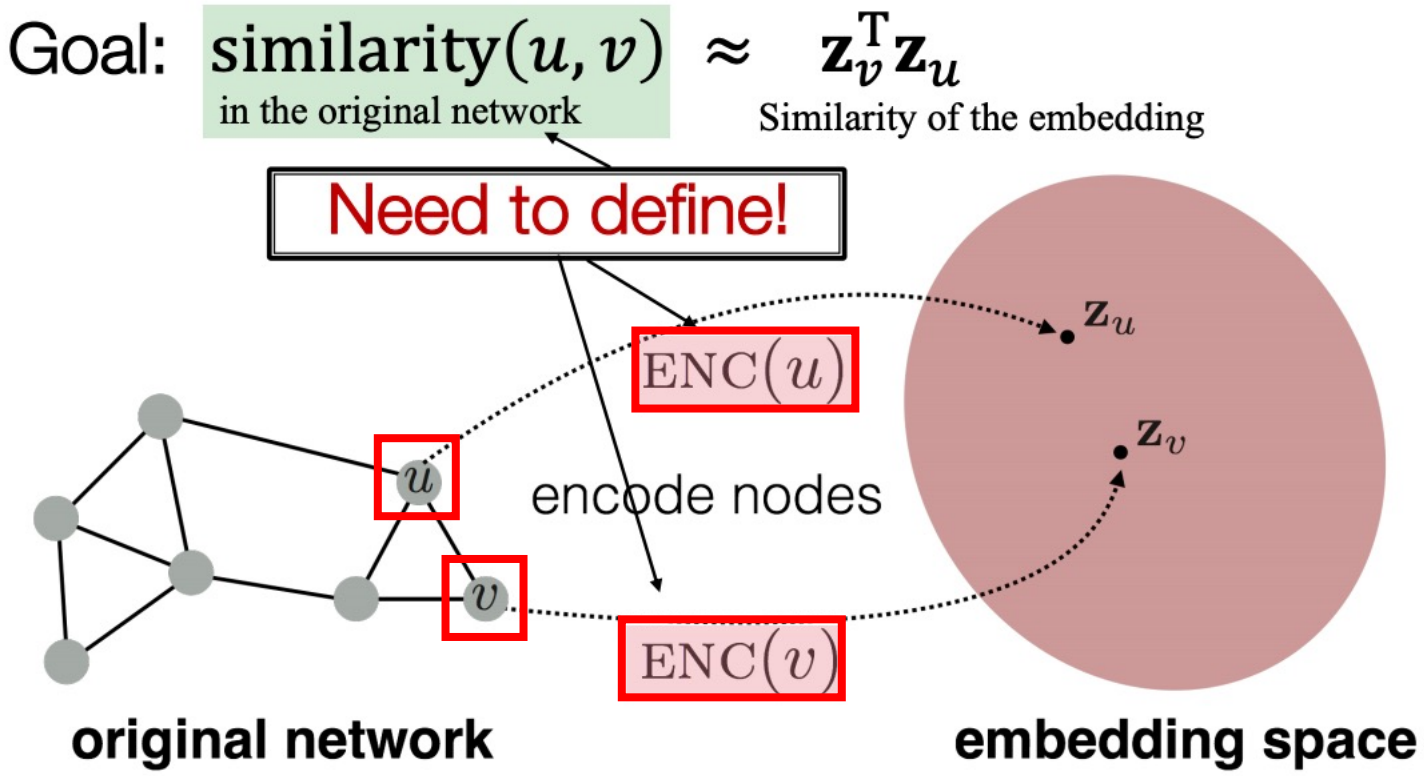


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data

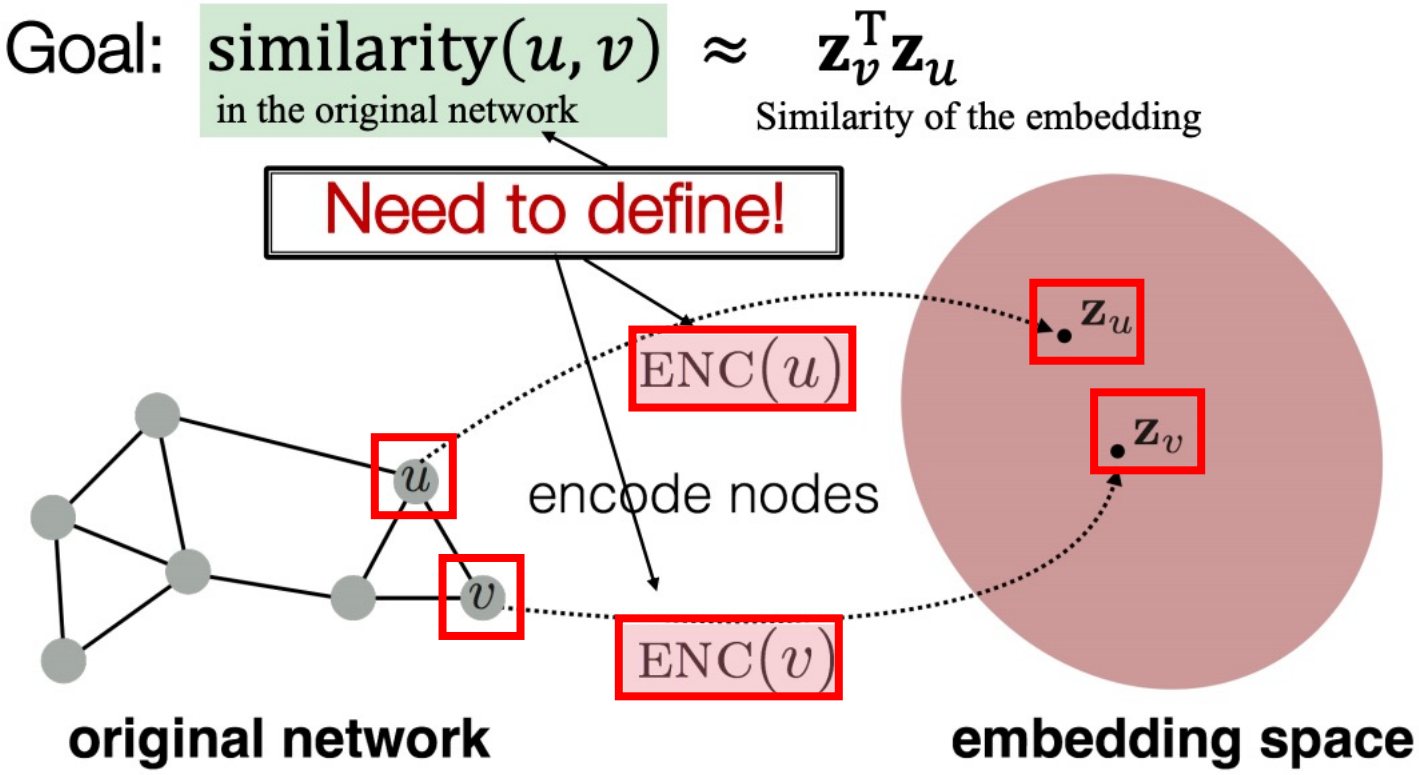


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data

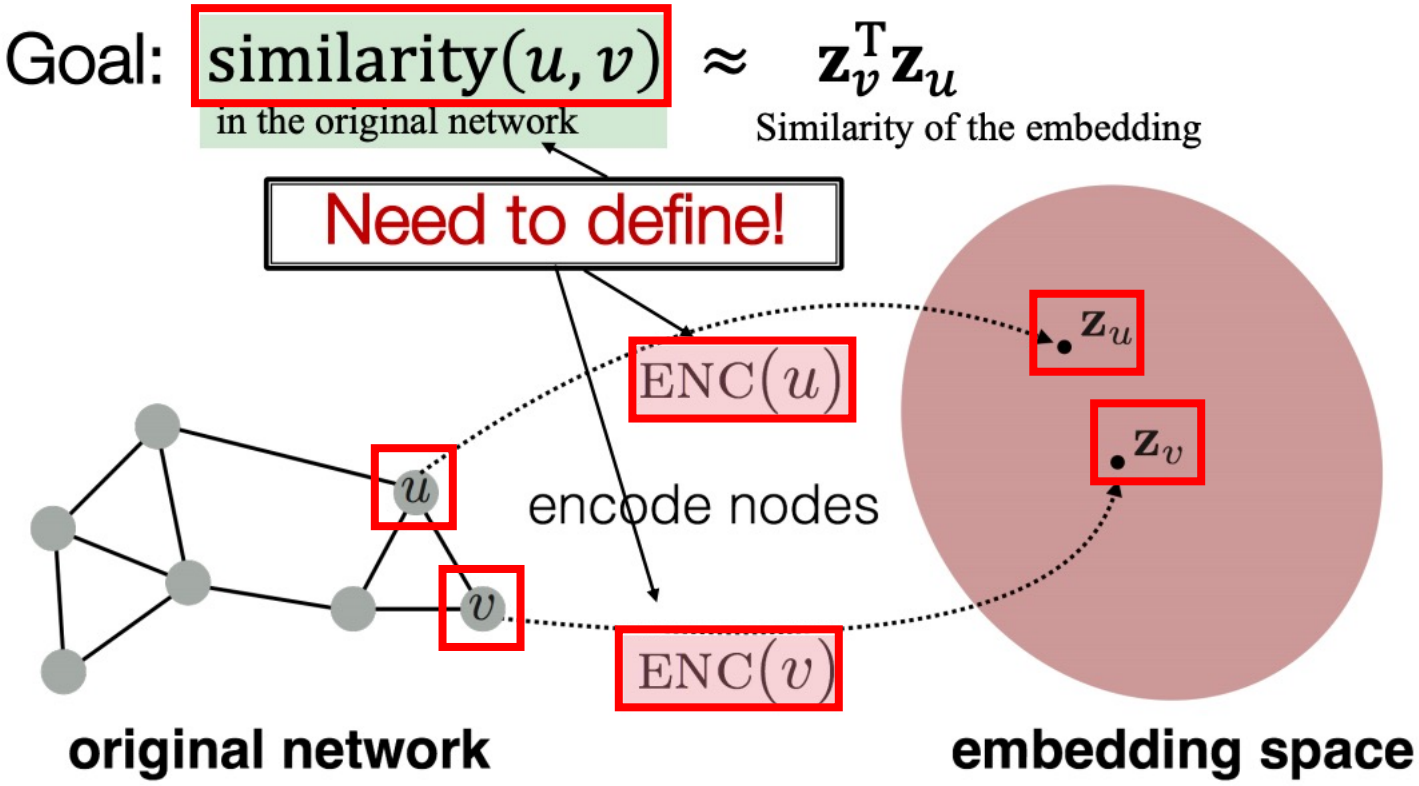


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data

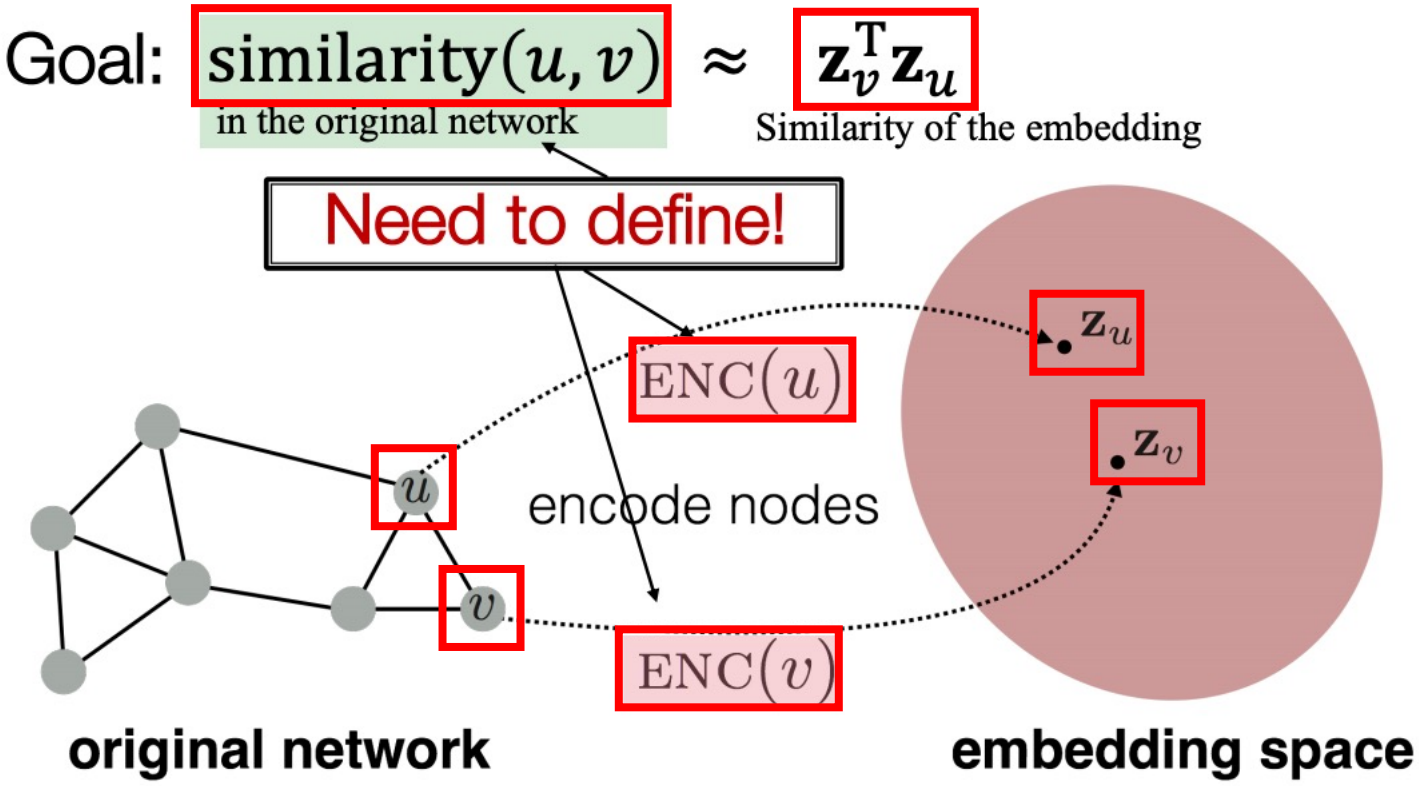


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data

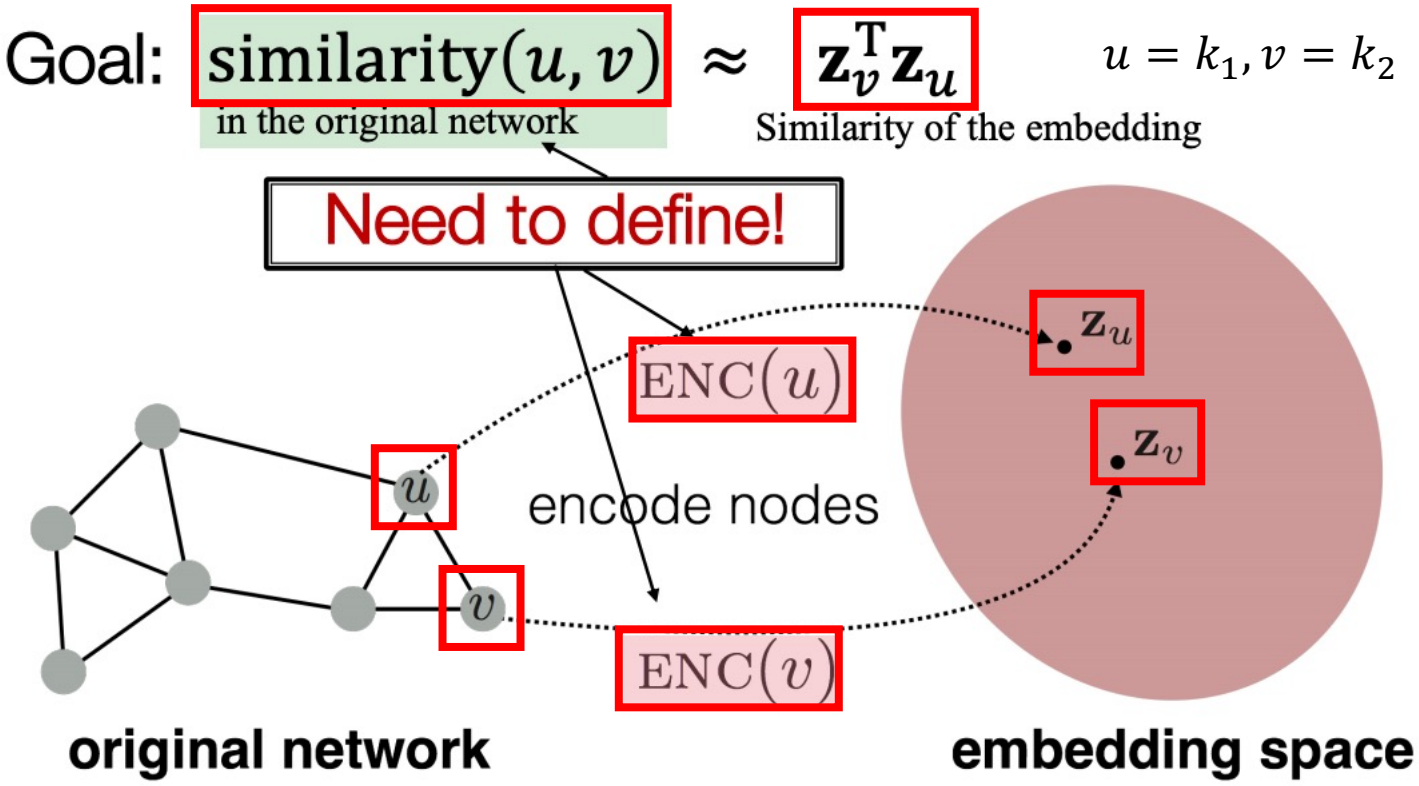


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data

Goal: $\text{similarity}(u, v) \approx \mathbf{z}_v^{\mathrm{T}} \mathbf{z}_u$ $\quad u = k_1, v = k_2$

similarity(u, v): in the original network

$\mathbf{z}_v^{\mathrm{T}} \mathbf{z}_u$: Similarity of the embedding

Q: how to learn ENC?

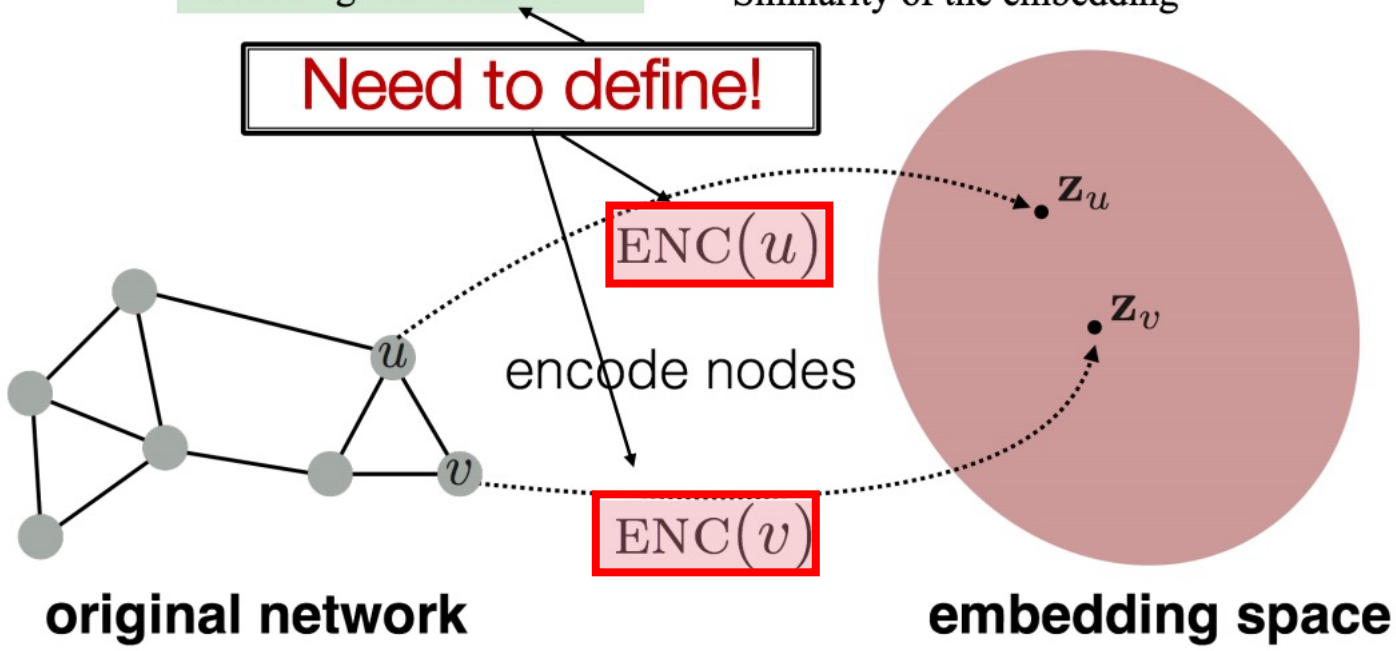
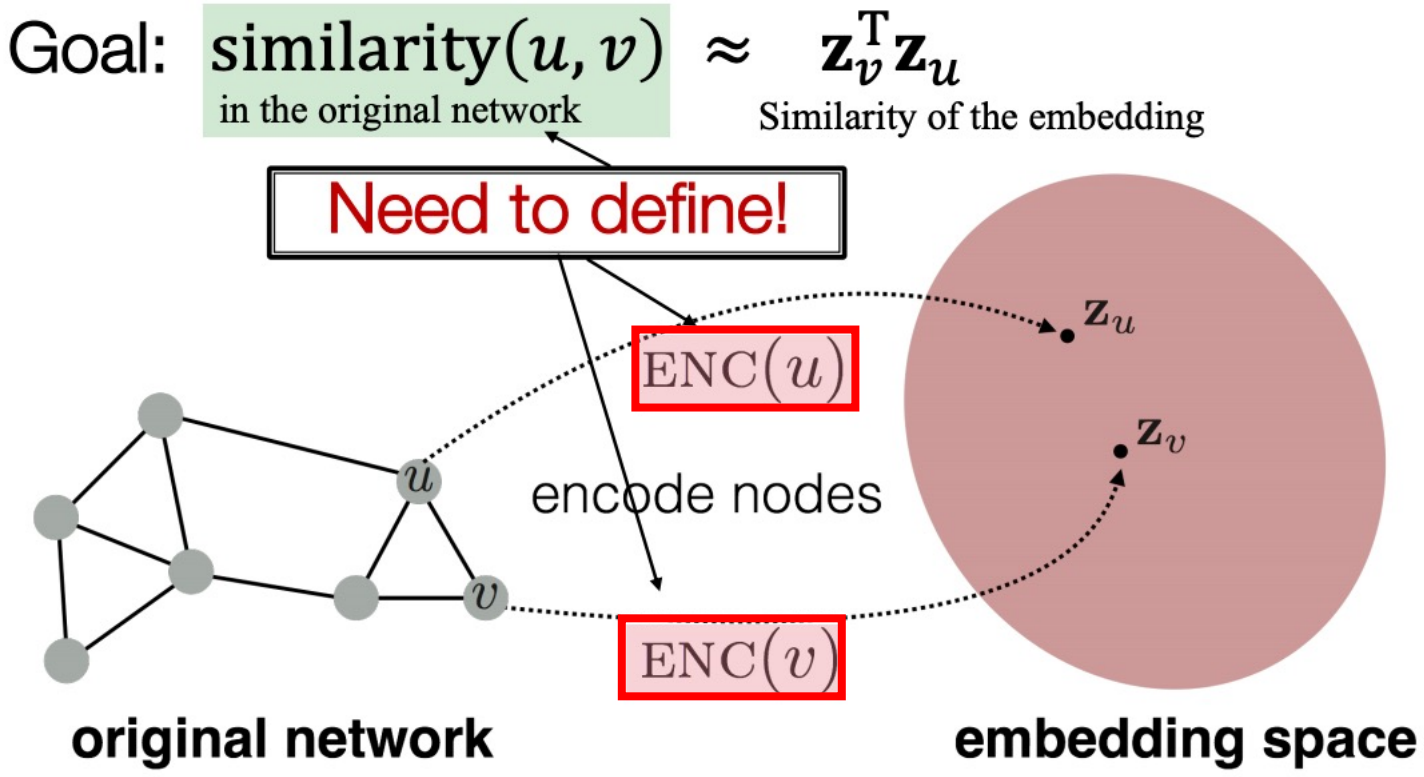


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?

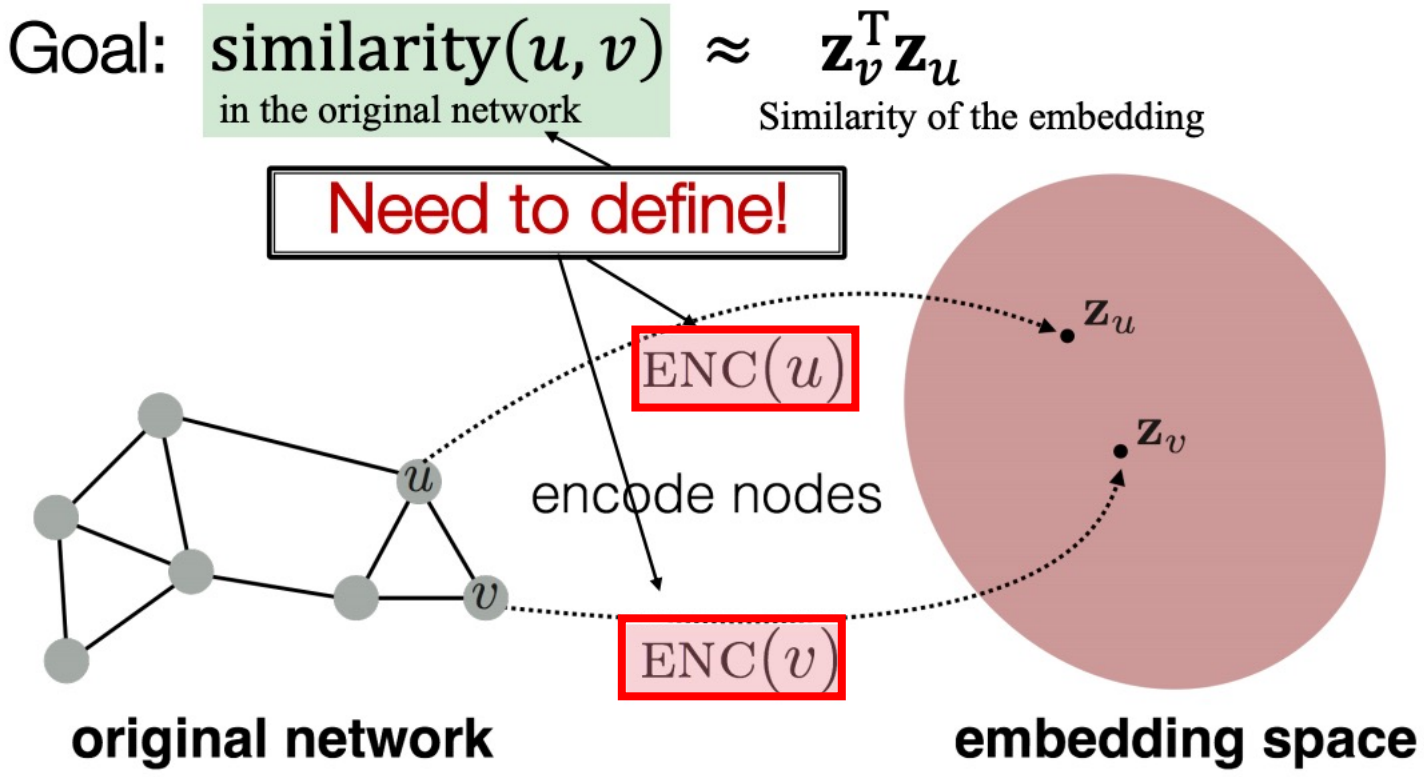
$$\boxed{\mathrm{ENC}(v)} = \mathbf{z}_v = \mathbf{Z} \cdot v$$

$\mathbf{Z} \in \mathbb{R}^{d \times |\mathcal{V}|}$ matrix, each column is a node embedding [what we learn / optimize]

$v \in \mathbb{I}^{|\mathcal{V}|}$ indicator vector, all zeroes except a one in column indicating node *v*

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?
Linear transformation

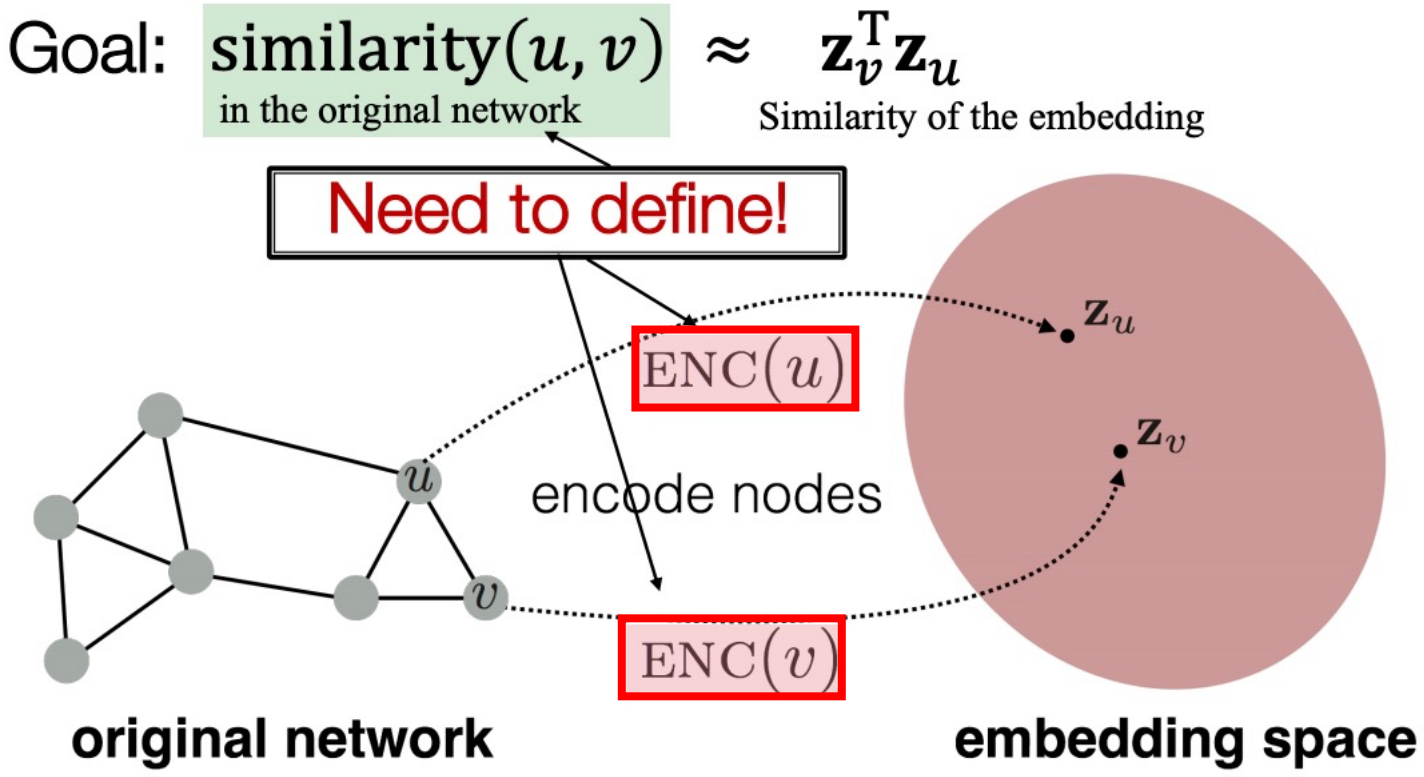
$$\mathrm{ENC}(v) = \mathbf{z}_v = \mathbf{Z} \cdot v$$

$\mathbf{Z} \in \mathbb{R}^{d \times |\mathcal{V}|}$ matrix, each column is a node embedding [what we learn / optimize]

$v \in \mathbb{I}^{|\mathcal{V}|}$ indicator vector, all zeroes except a one in column indicating node *v*

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?
Linear transformation

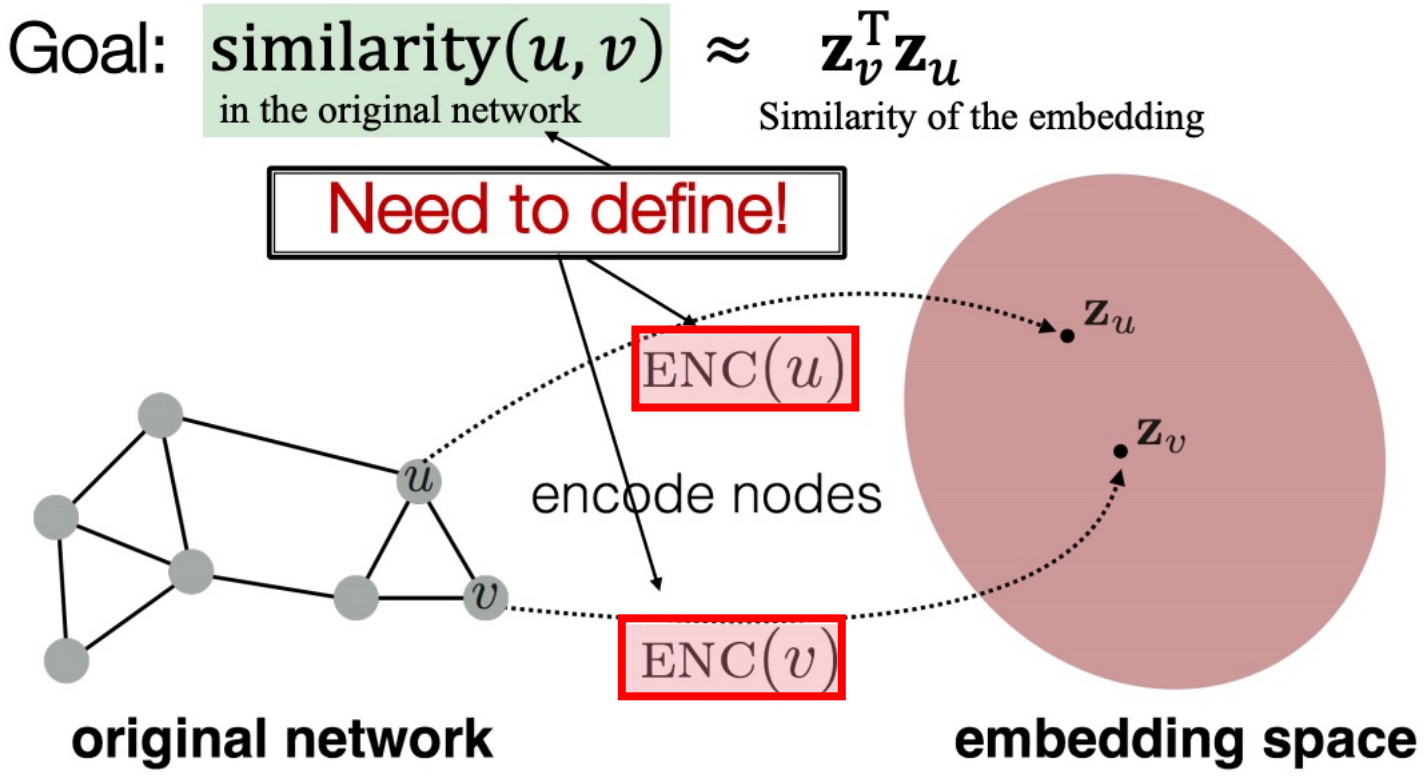
$$\mathrm{ENC}(v) = \mathbf{z}_v = \mathbf{Z} \cdot v$$

$\mathbf{Z} \in \mathbb{R}^{d \times |\mathcal{V}|}$ matrix, each column is a node embedding [what we learn / optimize]

$v \in \mathbb{I}^{|\mathcal{V}|}$ indicator vector, all zeroes except a one in column indicating node *v*

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?
Linear transformation

$$\mathrm{ENC}(v) = \mathbf{z}_v = \mathbf{Z} \cdot v$$

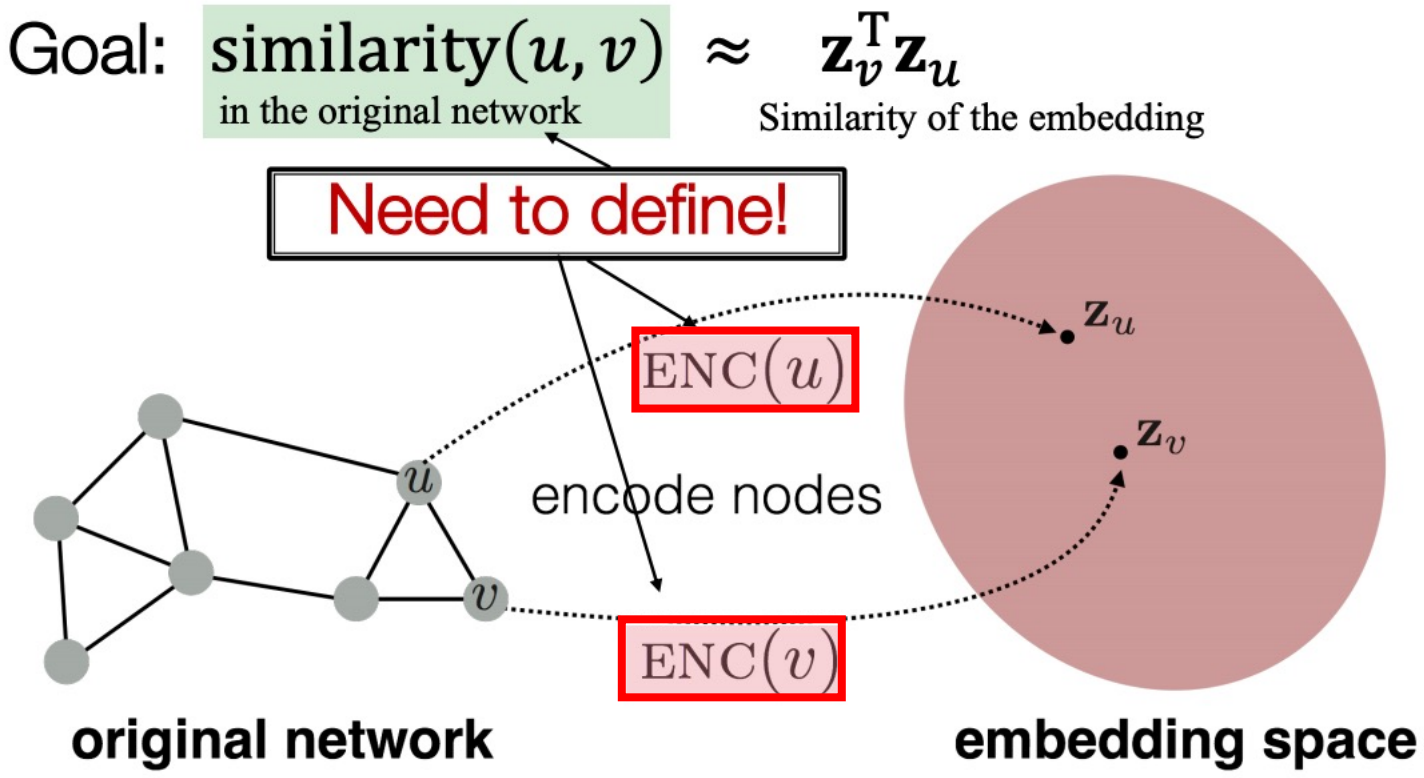
$\mathbf{Z} \in \mathbb{R}^{d \times |\mathcal{V}|}$ matrix, each column is a node embedding [what we learn / optimize]

$v \in \mathbb{I}^{|\mathcal{V}|}$ indicator vector, all zeroes except a one in column indicating node *v*

One-hot labels

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?
Linear transformation

$\mathrm{ENC}(v) = \mathbf{z}_v = \mathbf{Z} \cdot v$

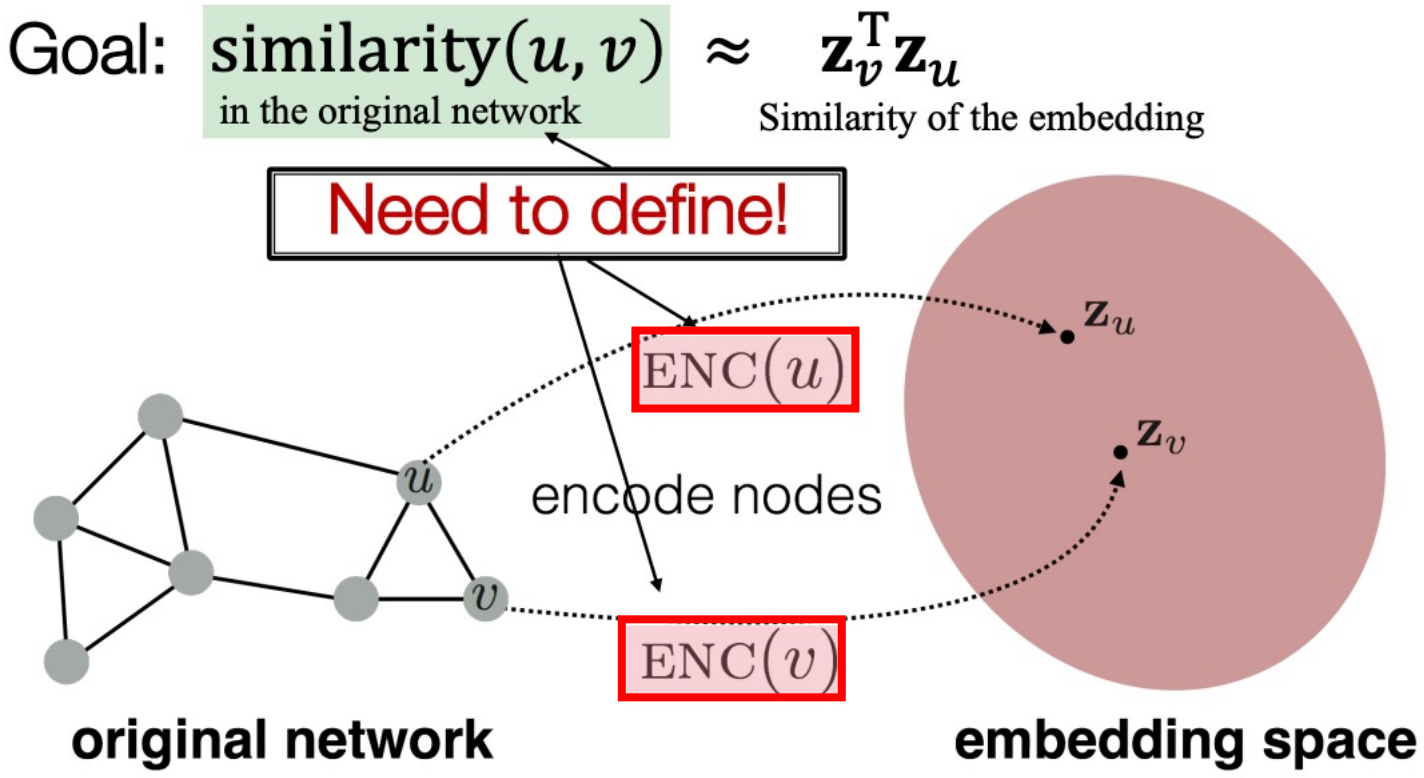
$\mathbf{Z} \in \mathbb{R}^{d \times |\mathcal{V}|}$

$v \in \mathbb{I}^{|\mathcal{V}|}$

Binary	Gray code	One-hot
000	000	00000001
001	001	00000010
010	011	00000100
011	010	00001000
100	110	00010000
101	111	00100000
110	101	01000000
111	100	10000000

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?

Linear transformation

$\mathrm{ENC}(v) = \mathbf{z}_v = \mathbf{Z} \cdot v$

$\mathbf{Z} \in \mathbb{R}^{d \times |\mathcal{V}|}$

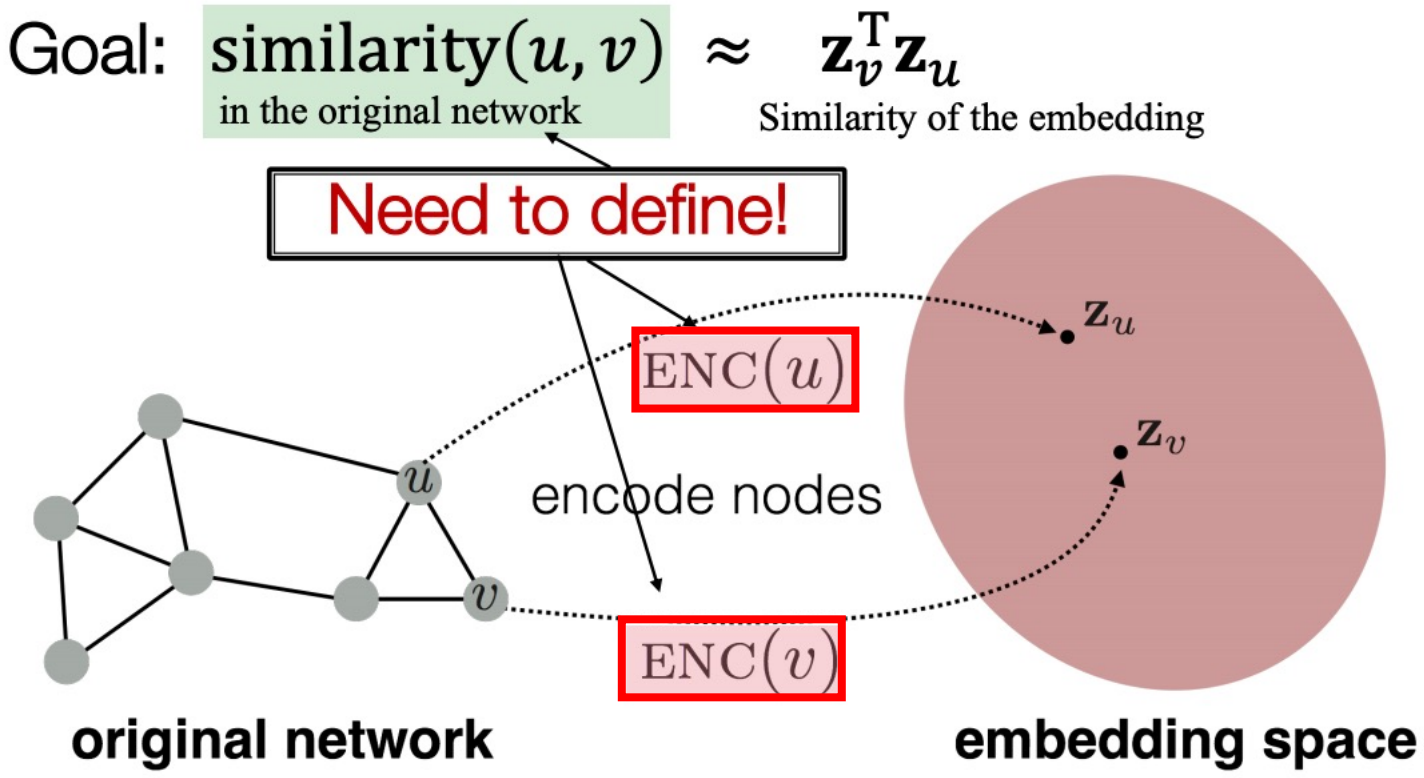
$v \in \mathbb{I}^{|\mathcal{V}|}$

Each node

Binary	Gray code	One-hot
000	000	00000001
001	001	00000010
010	011	00000100
011	010	00001000
100	110	00010000
101	111	00100000
110	101	01000000
111	100	10000000

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?
Linear transformation

$\mathrm{ENC}(v) = \mathbf{z}_v = \mathbf{Z} \cdot v$

$\mathbf{Z} \in \mathbb{R}^{d \times |\mathcal{V}|}$

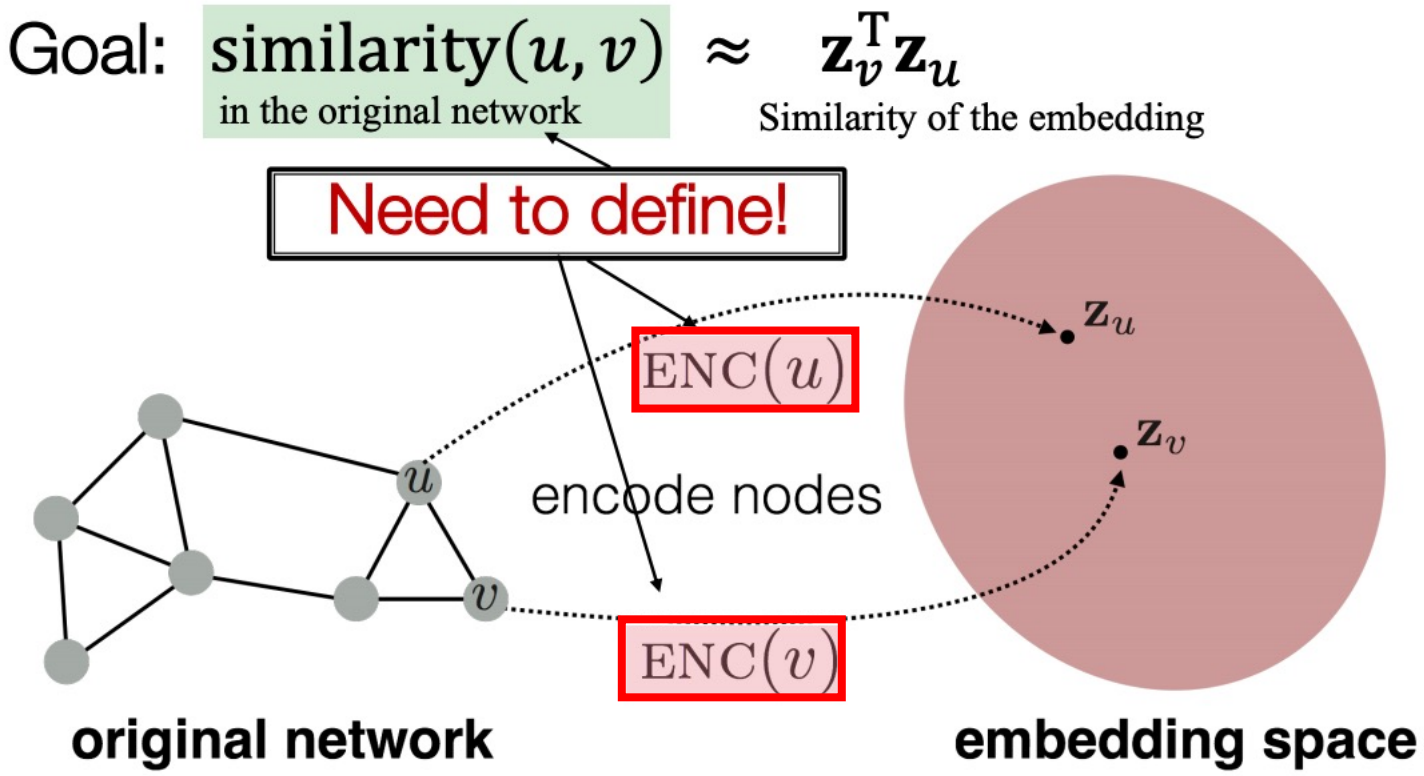
$v \in \mathbb{I}^{|\mathcal{V}|}$

Each node

Binary	Gray code	One-hot
000	000	00000001
001	001	00000010
010	011	00000100
011	010	00001000
100	110	00010000
101	111	00100000
110	101	01000000
111	100	10000000

Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?

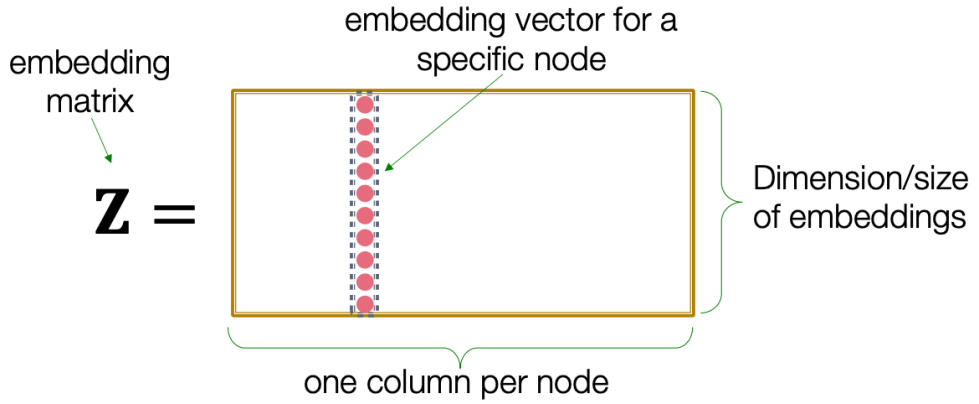
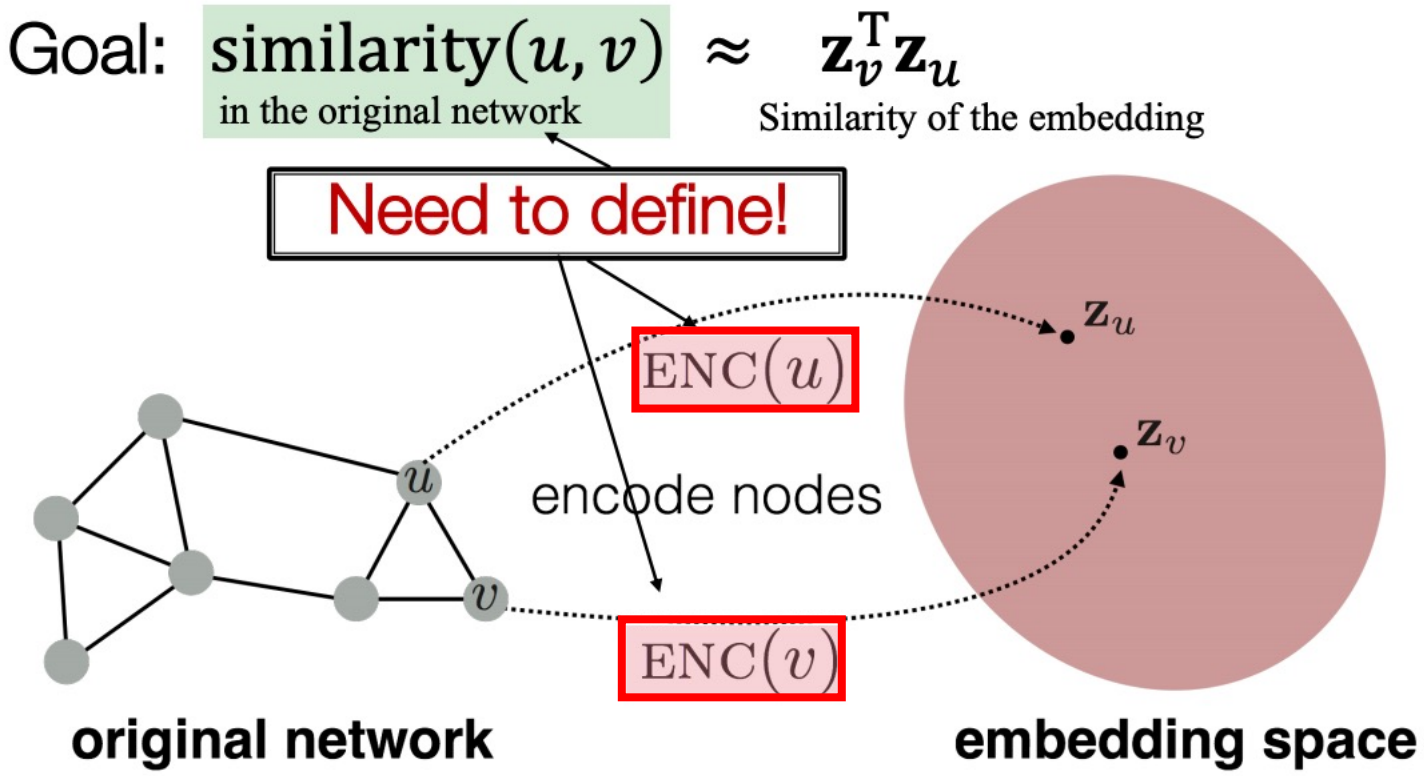


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?

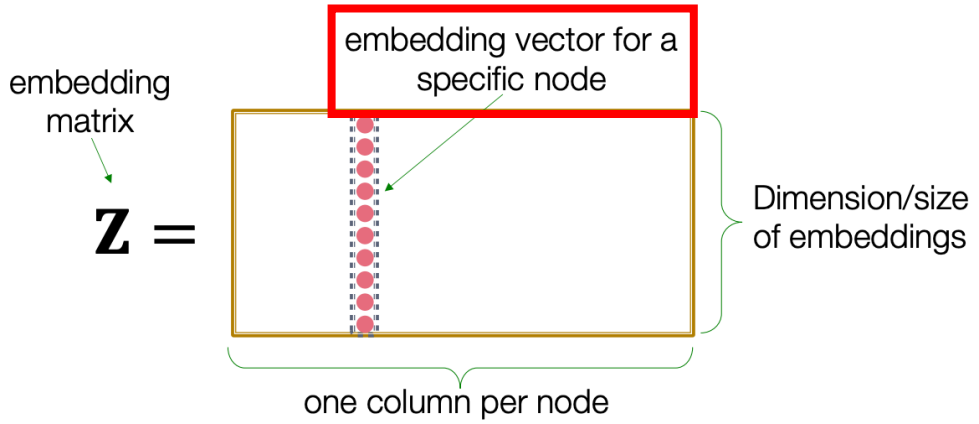
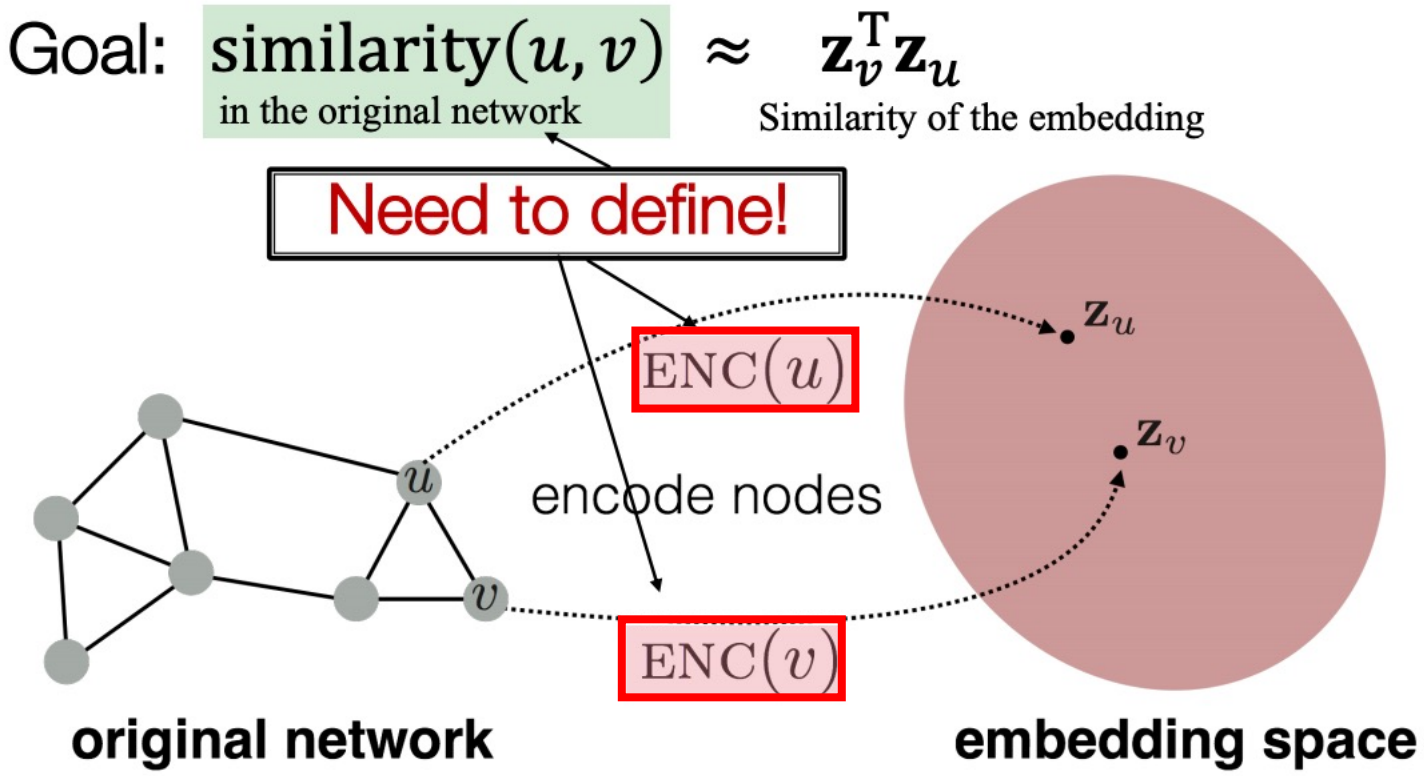


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?

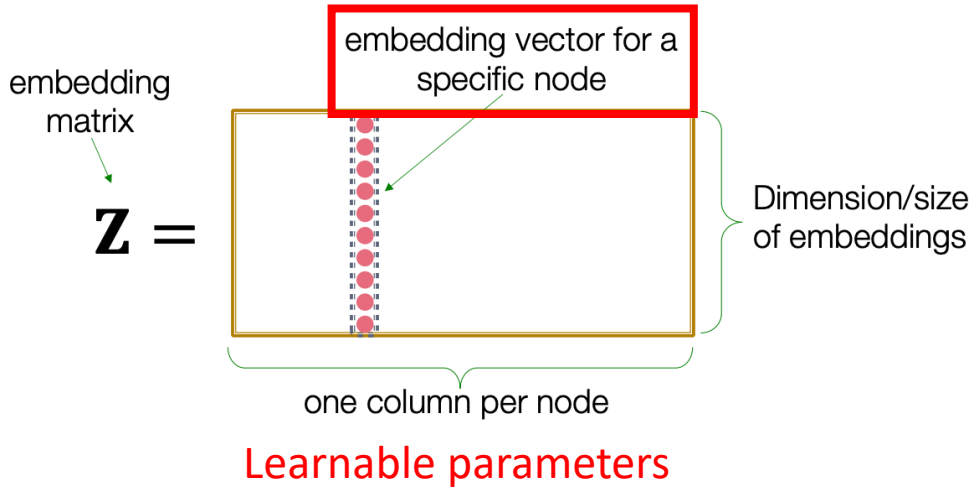
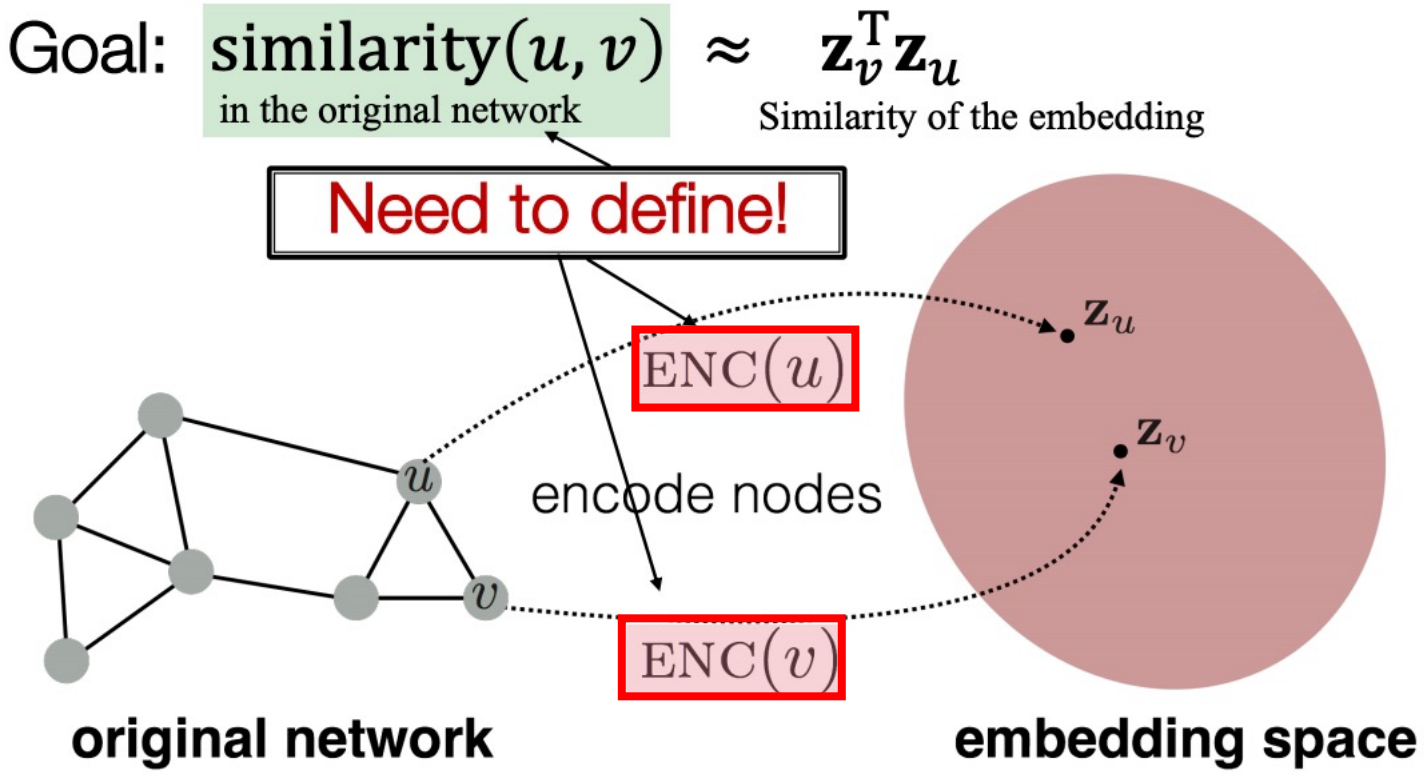
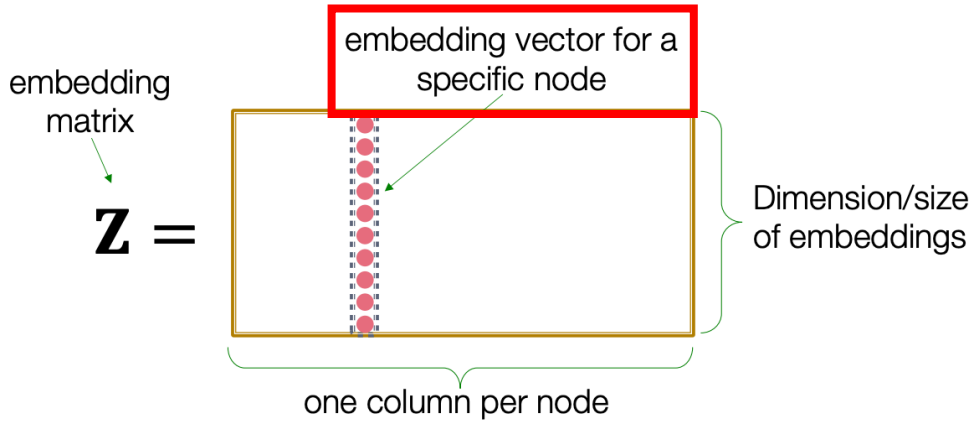


Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Encoder-decoder for graph data



Q: how to learn ENC?

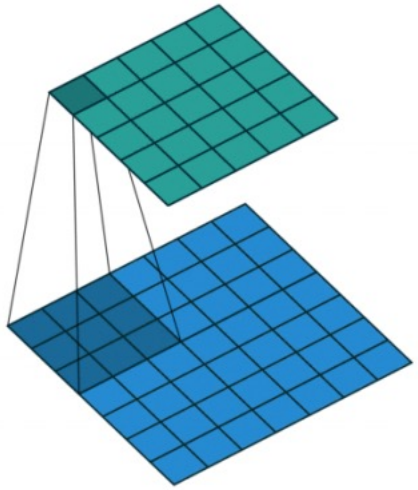


Learnable parameters

Q: how can we replace linear node embedding with nonlinear functions?

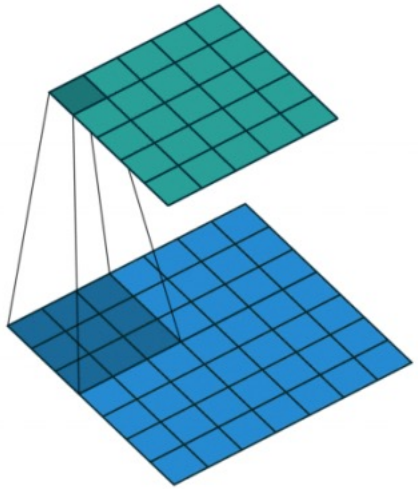
Image credit http://web.stanford.edu/class/cs224w/slides/03-nodeemb.pdf

Graph neural networks



Image

Graph neural networks



Q: can we use convolution operation on graph?

Graph convolutional neural networks