

# Graph Theory and Applications



## Lecture 9: Deep Graph Learning

Slide adapt from <https://ai.tencent.com/ailab/ml/KDD-Deep-Graph-Learning.html>

Ta Viet Cuong, Ph.D  
HMI laboratory, FIT-UET

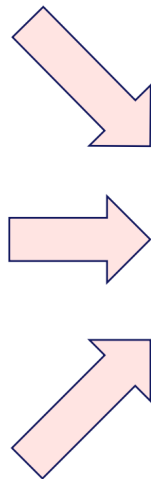
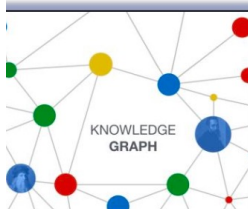
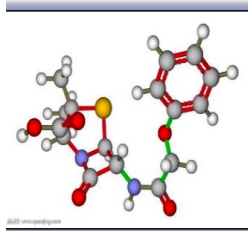
# Today

I. Graph Neural Network

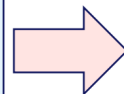
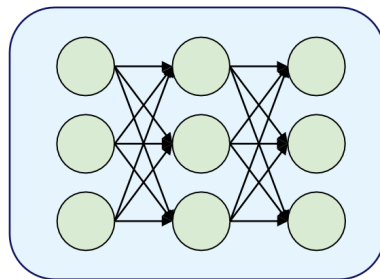
II. Một số cấu trúc GNN cơ bản

III. Các bài toán mở

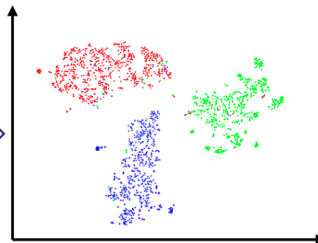
# Graph Neural Networks



Graph Neural Network



Graph/Node Representation



## Applications

Node Classification

Link Prediction

Community Detection

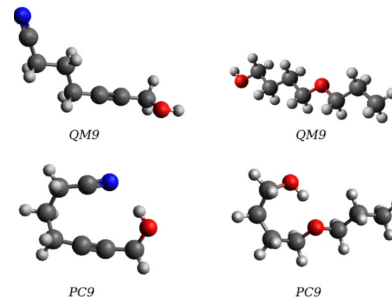
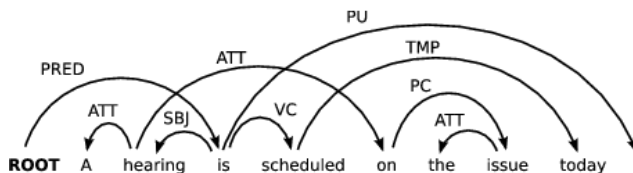
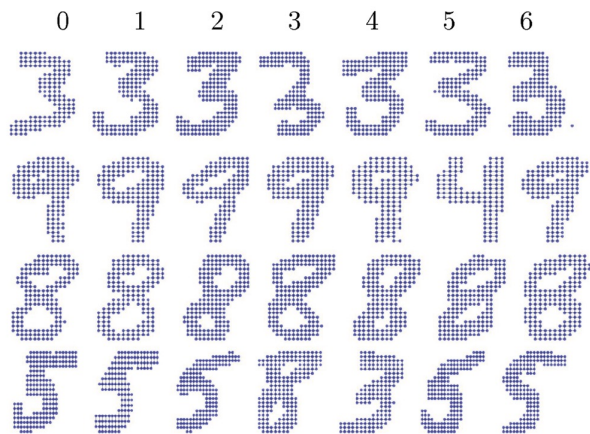
Graph Generation

.....

# Một số ứng dụng

GNNs là một nhánh quan trọng của học máy trên đồ thị:

- Dữ liệu đồ thị ngày càng phát triển
- Ảnh và Văn bản cũng có thể tính là một dạng đồ thị đặc biệt



# Dữ liệu trên đồ thị

GNNs là một nhánh quan trọng của học máy trên đồ thị:  $G = \langle V, E, X \rangle$

Trong đó:

- $V$ : Tập đỉnh
- $E$ : Tập cạnh (có thể bao gồm cả thuộc tính)
- $X$ : Đặc trưng của mỗi đỉnh

Mục tiêu:

- Học hàm  $f_\theta: V \rightarrow Y$  với  $Y$  là nhãn mục tiêu cần học.

# Ví dụ thực tế:

Bài toán dự đoán cạnh trên mạng xã hội

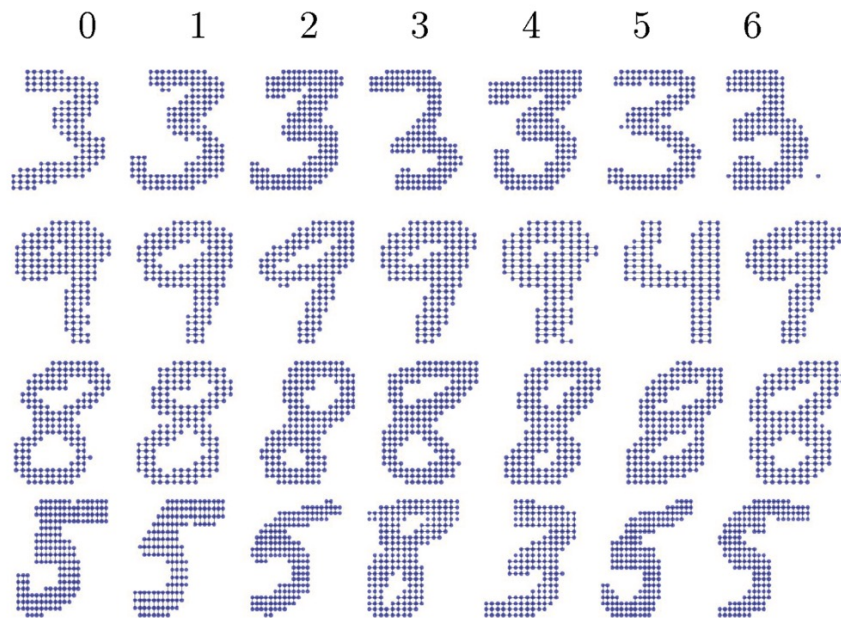
- **V**: Tập đỉnh là người dùng của mạng xã hội
- **E**: Tập cạnh hiện tại, đi kèm thuộc tính có thể là: có friends hay ko có friends/lần chat gần nhất v..v...
- **X**: Đặc trưng của mỗi đỉnh

Mục tiêu:

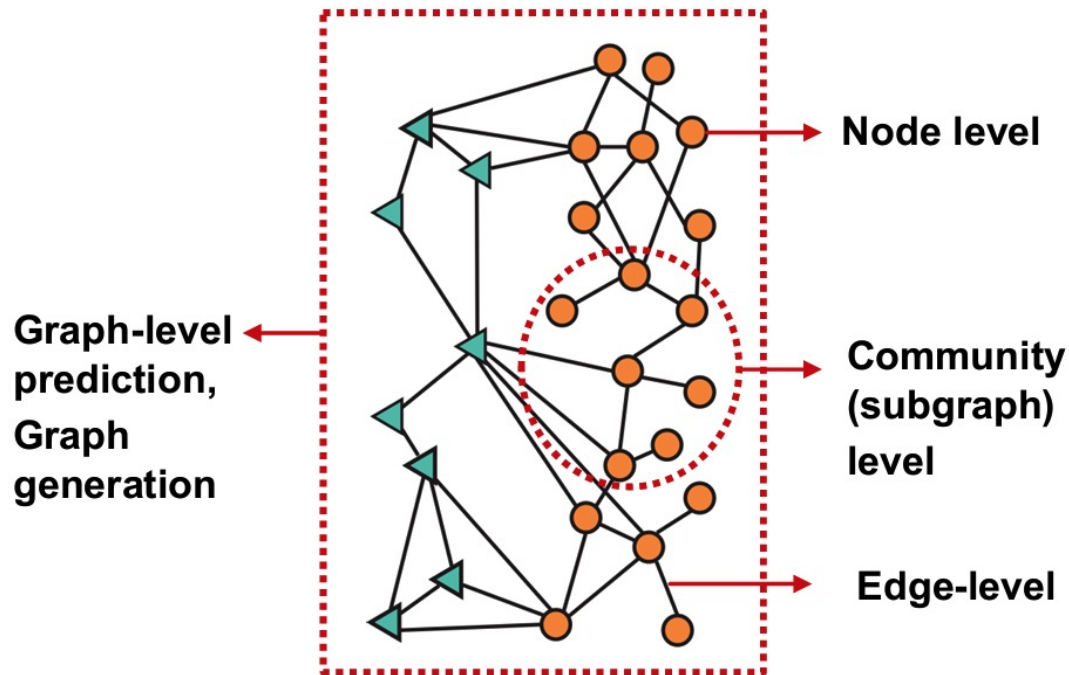
- Học hàm  $f_\theta: V \times V \rightarrow \{0, 1\}$  Dự đoán xem hai người có là bạn bè

# Ví dụ thực tế: MNIST

Bài toán phân loại ảnh

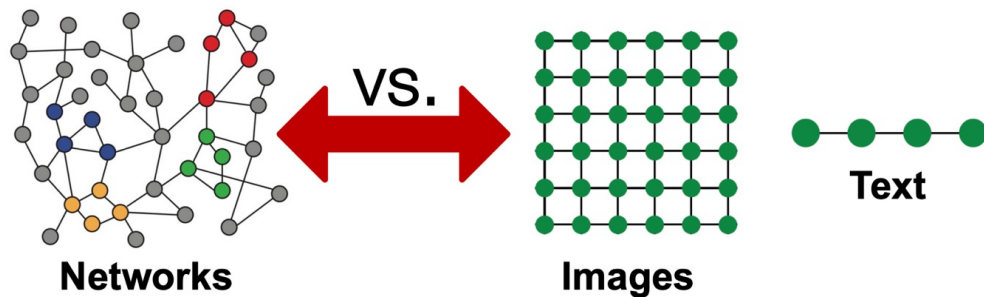


# Một số bài toán học máy cơ bản



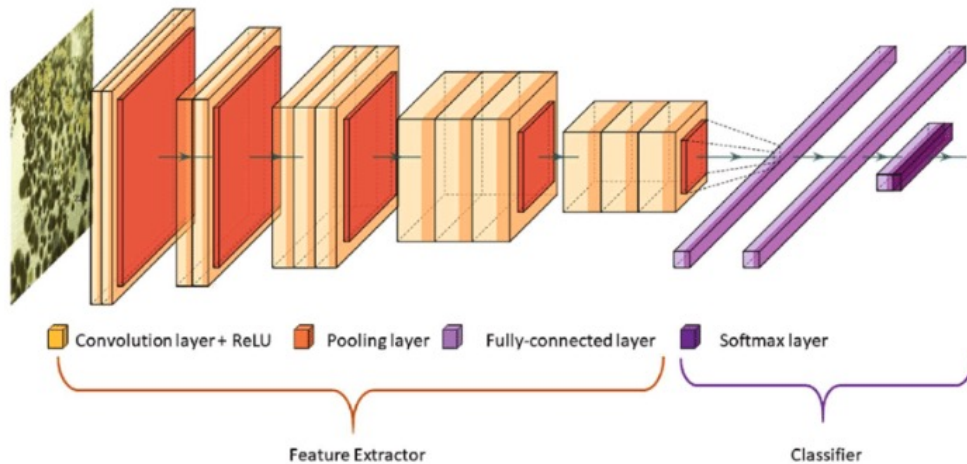


# GNN



- Dữ liệu đồ thị là dạng tổng quát
- Từ 2012-2015: Mạng học sâu nhân chập xuất hiện và giải quyết được phần lớn các bài toán ảnh
- 2015-bây giờ: Các mô hình học máy hồi qui RNN/LSTM sau đó là Transformer, giải quyết được các bài toán của văn bản
- Mô hình tương tự cho đồ thị vẫn chưa xuất hiện

# Cách tiếp cận của học sâu



- Mô hình hoá hàm mục tiêu bằng tham số  $\theta$ :

$$f_{\theta}: V \rightarrow Y$$

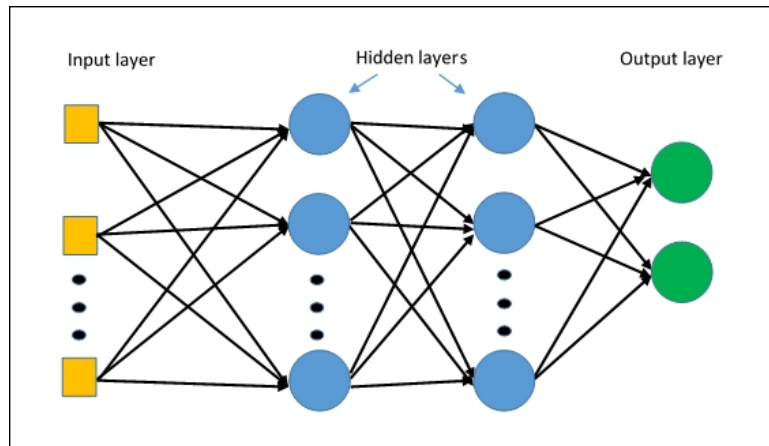
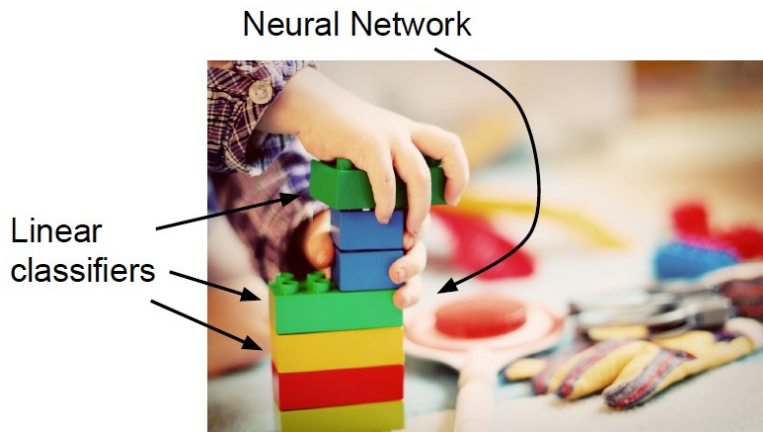
Có thể huấn luyện bằng gradient descent

# Cách tiếp cận của học sâu

Mô hình hoá hàm mục tiêu bằng tham số  $\theta$ :

$$f_{\theta}: V \rightarrow Y$$

Có thể huấn luyện bằng gradient descent

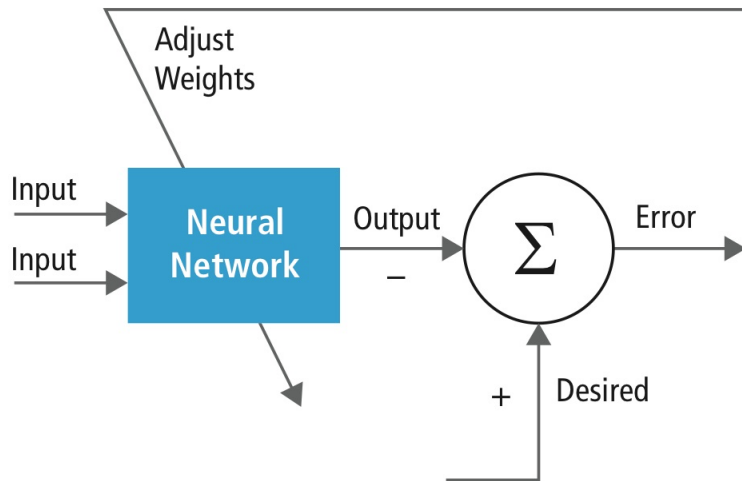


# Cách tiếp cận của học sâu

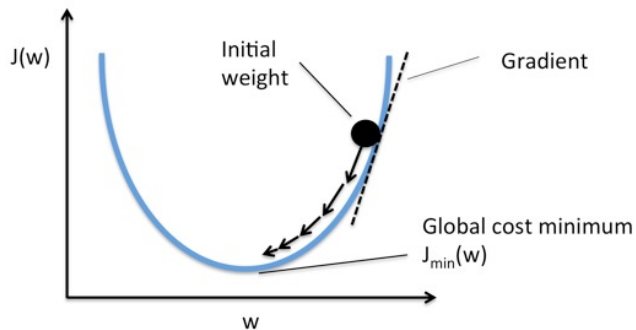
Mô hình hoá hàm mục tiêu bằng tham số  $\theta$ :

$$f_{\theta}: V \rightarrow Y$$

Có thể huấn luyện bằng gradient descent



# Cách tiếp cận của học sâu

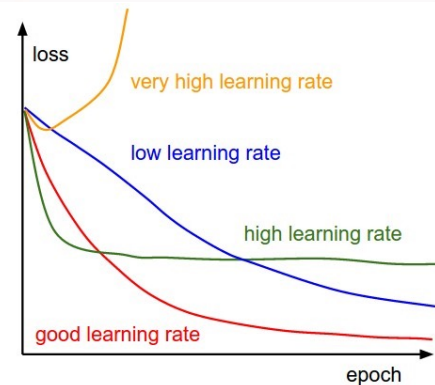
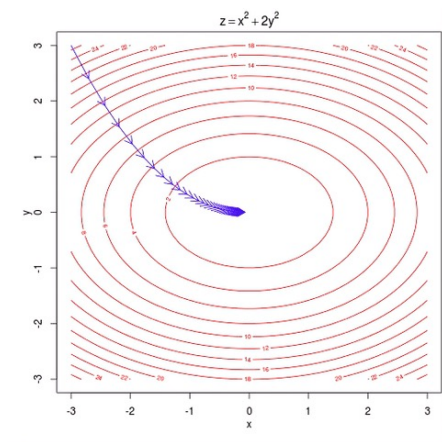


**Algorithm:** Gradient Descent

**Input:**  $Y, \Theta, \mathbf{X}, \alpha$ , tolerance, max iterations

**Output:**  $\Theta$

```
1 for  $i = 0; i < \text{max iterations}; i++$  do
2   current cost =  $\text{Cost}(Y, \mathbf{X}, \Theta)$ 
3   if current cost < tolerance then
4     break
5   else
6     gradient =  $\text{Gradient}(Y, \mathbf{X}, \Theta)$ 
7      $\theta_j \leftarrow \theta_j - \alpha \cdot \text{gradient}$ 
```



# Cách tiếp cận của học sâu

Mô hình hoá hàm mục tiêu bằng tham số  $\theta$ :

$$f_{\theta}: V \rightarrow Y$$

Có thể huấn luyện bằng gradient descent

Vấn đề với GNN:

- Tích hợp thông tin cấu trúc vào  $\theta$
- Cần bao nhiêu dữ liệu để học thông tin cấu trúc
- Có thể tổng quát hoá được không?

# Today

I. Graph Neural Network

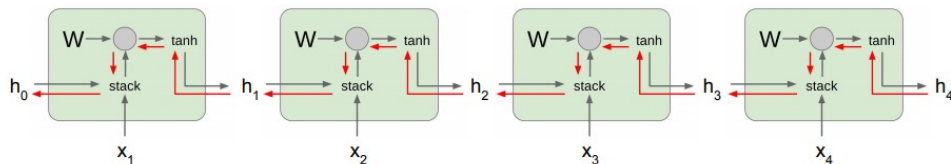
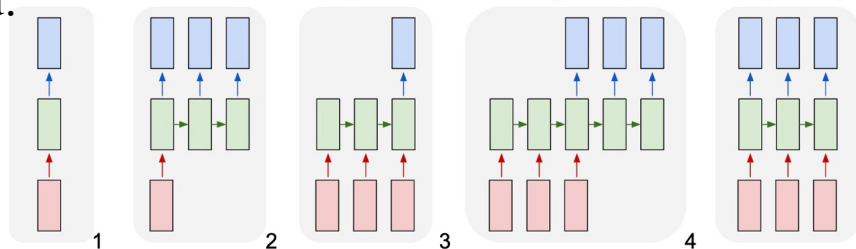
II. Một số cấu trúc GNN cơ bản

III. Các bài toán mở

# GNN có thể xem như là RNN

Bài toán mô hình hoá chuỗi với mạng học sâu:

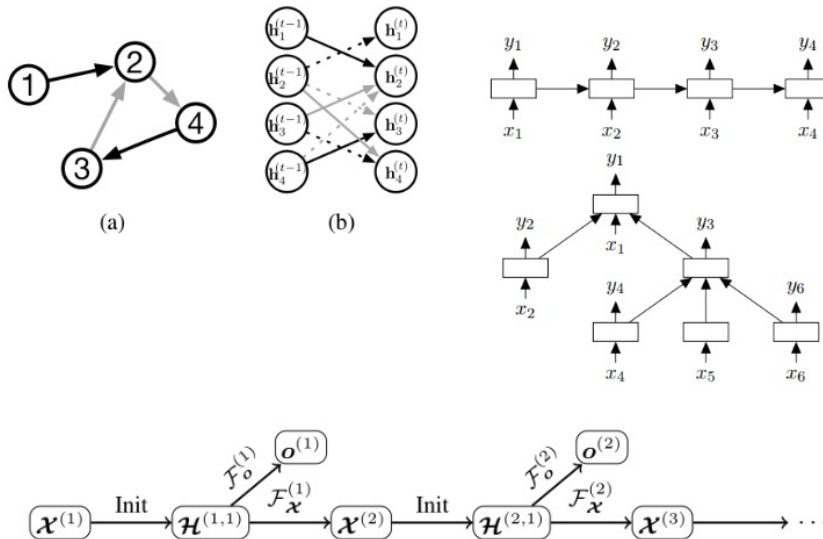
- Dự đoán câu kế tiếp
- Sinh câu kế tiếp
- Phân loại câu
- Sinh chú thích cho ảnh
- Sinh code (?)
- Làm toán (?)





# GNN có thể xem như là RNN

- Các random walk đưa thông tin về local, có thể mô hình hoá thành các câu
- Xem các đỉnh như là các từ



# GNN có thể xem như là mạng nhân chập

Identity



0	0	0
0	1	0
0	0	0

Blur

1	1	1
1	1	1
1	1	1

Sharpen

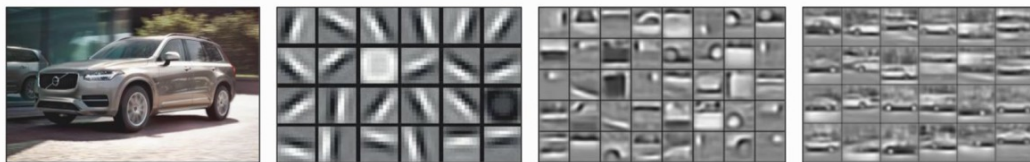
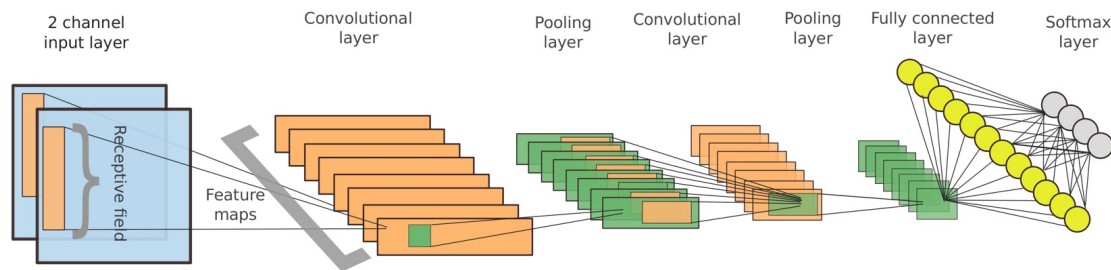
0	-1	0
-1	4	-1
0	-1	0

Edge detector

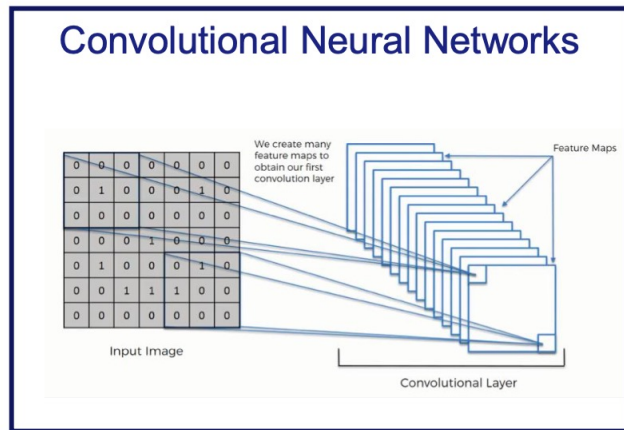
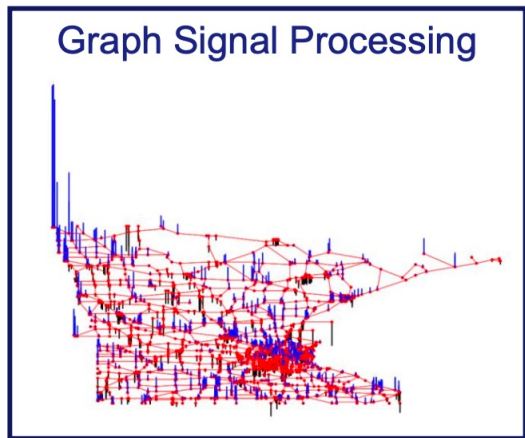
0	0	0
-1	1	0
0	0	0



# GNN có thể xem như là mạng nhân chập

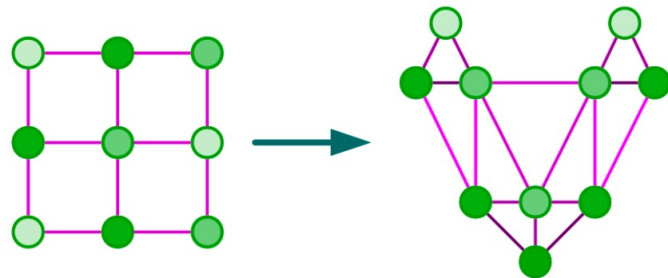


# GNN có thể xem như là mạng nhân chập



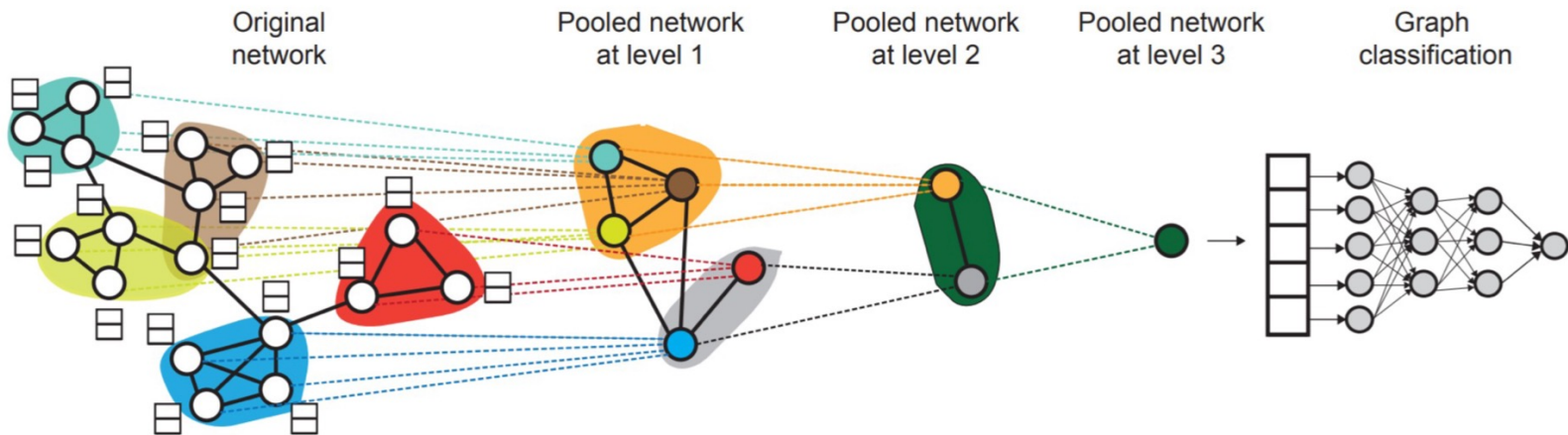
How to perform the convolution on graphs?

- Irregular structures.
- Weighted edges.
- No orientation or ordering (in general).



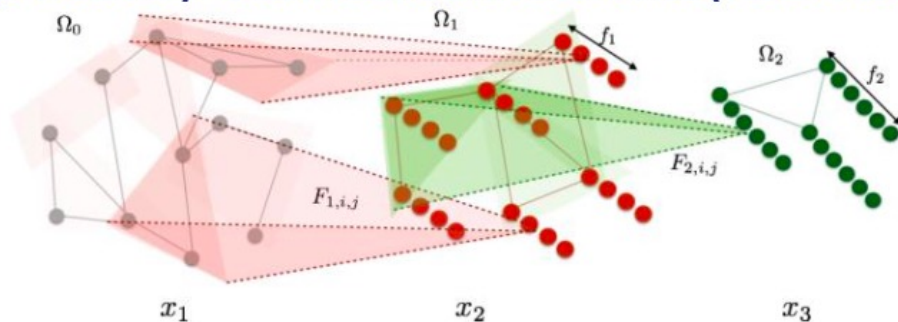
# GNN có thể xem như là mạng nhân chập

Tổng hợp và lọc thông tin:



# GNN có thể xem như là mạng nhân chập

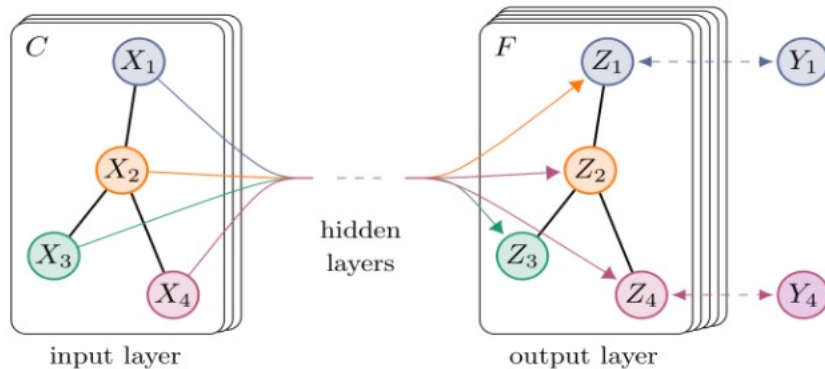
Deep Locally Connected Networks(ICLR 2014) [1]



- Discuss two constructions on both spatial and spectral domain.
- Analog the convolution operation based on the Laplacian spectrum.
- **Additional eigen decomposition is needed.**

# GNN có thể xem như là mạng nhân chập

## Graph Convolutional Network (ICLR 2017)



$$\mathbf{H}^{(l+1)} = \sigma(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(l)} \mathbf{W}^{(l)})$$

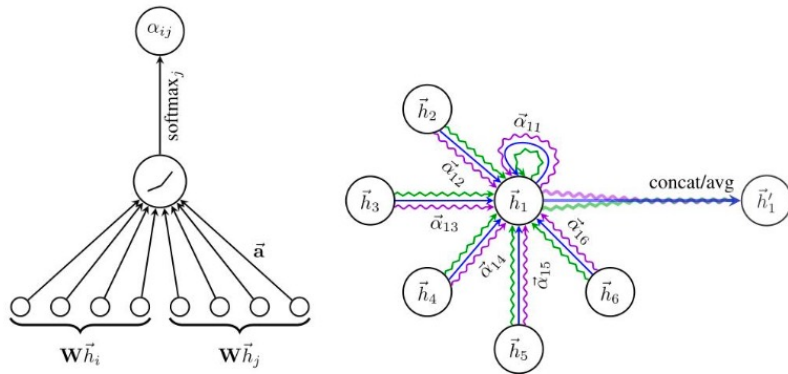
Approximate 1-order Chebyshev polynomial the in spatial domain.  
Layer-wise convolution to extend receptive field.

**The practical convolutional model for graphs.**

# GNN có thể xem như là mạng nhân chập

Tổng hợp và lọc thông tin:

- Sử dụng attention score như trong mạng biến đổi
- Các đỉnh giống nhau thì sẽ có nhiều ảnh hưởng hơn



Replace the fixed aggregation weight  $a_{ij}$  to the learnable self-attention.

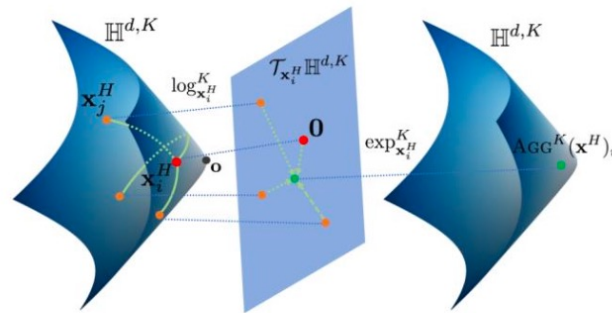
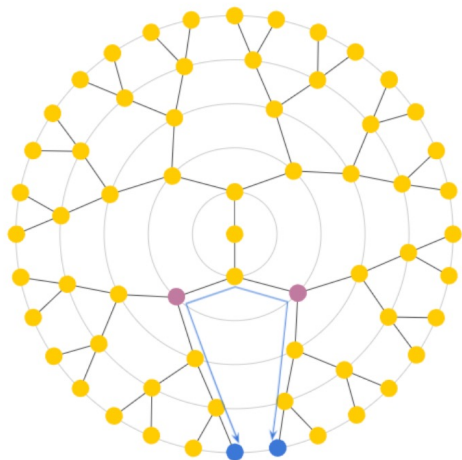
$$\mathbf{h}_i^{(l+1)} = \sigma \left( \sum_{j \in N(v_i)} \mathbf{a}_{ij} W^{(l)} \mathbf{h}_j^{(l)} \right)$$
$$\mathbf{a}_{ij} = \exp \left( \frac{\sigma \left( \boldsymbol{\alpha}^T [\mathbf{W} \mathbf{h}_i || \mathbf{W} \mathbf{h}_j] \right)}{\sum_{k \in N(v_i)} \boldsymbol{\alpha}^T [\mathbf{W} \mathbf{h}_i || \mathbf{W} \mathbf{h}_k]} \right)$$



# GNN có thể xem như là mạng nhân chập

Tổng hợp và lọc thông tin:

- Dữ liệu trên đồ thị thường có tính cong nhất định
- Sử dụng mạng nhân chập trong miền Hyperbolic



Construct the GCN in hyperbolic space.

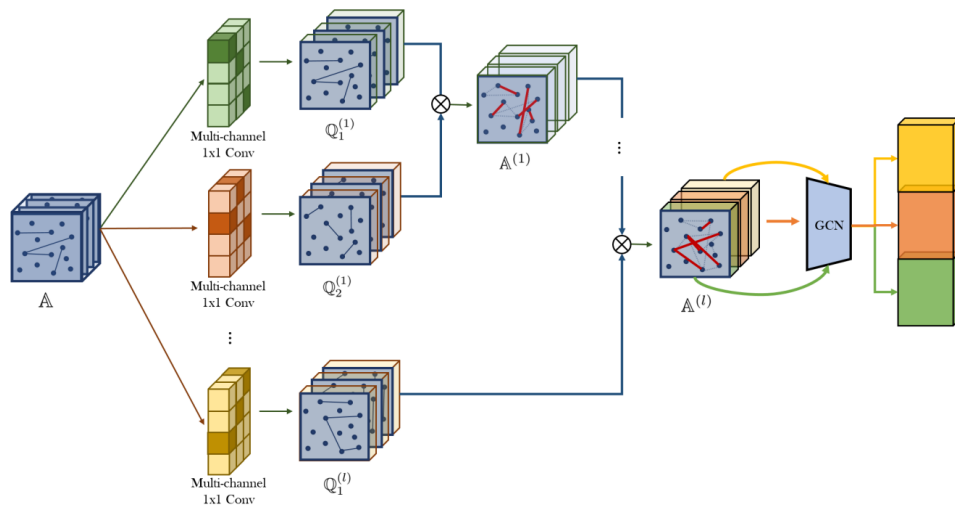
- Smaller distortion.
- Suitable for scale-free and hierarchical structure.
- Hyperbolic feature transform.  
$$\mathbf{h}_i^{(l+1),H} = (\mathbf{W}^{(l+1)} \otimes_{K_l} \mathbf{h}_i^{(l),H}) \oplus_{K_l} \mathbf{b}^{(l+1)}$$
- Attention-based hyperbolic aggregation.

$$\mathbf{y}_i^{(l+1),H} = \text{AGG}^{K_l}(\mathbf{h}^{(l),H})_i$$

# GNN với Transformer

Có một số biến thể của Transformer cho đồ thị:

- Mã hoá cấu trúc
- Lan truyền thông tin



# Cài đặt GNN

- Message Passing Framework:

- **Step 1:** Gather and transform the messages from neighbors:

$$\mathbf{m}_i^{(l+1)} = \text{AGG} \left( \{ M^{(l+1)}(\mathbf{h}_i^{(l)}, \mathbf{h}_j^{(l)}, \mathbf{e}_{i,j}) \mid j \in N(v_i) \} \right)$$

The message generation function.

**Input:** the state of current node, the state of the neighbor node and the edge features.

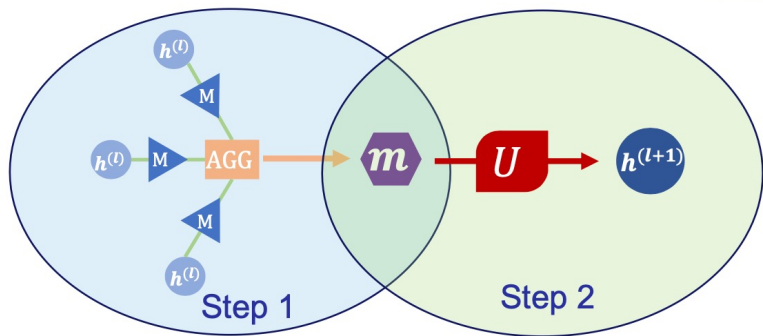
The neighborhood set of node.  
E.g. 1-hop neighbors.

- **Step 2:** Update the state of the target node.

$$\mathbf{h}_i^{(l+1)} = U^{(l+1)}(\mathbf{h}_i^{(l)}, \mathbf{m}_i^{(l+1)})$$

The aggregation function.  
E.g. SUM/MEAN/LSTM

The state update function.



- Most of current spatial GNNs can be formulated as a message passing process.

# Today

I. Graph Neural Network

II. Một số cấu trúc GNN cơ bản

III. Các bài toán mở

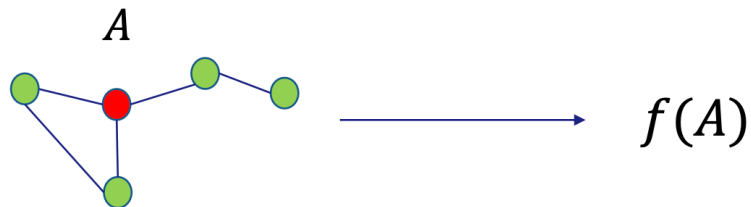
# Một số bài toán mở

Given any two graphs, can GNN determine if they are isomorphic or not?



# Một số bài toán mở

For any function on graphs, if there is a GNN approximating it up to an arbitrary accuracy?

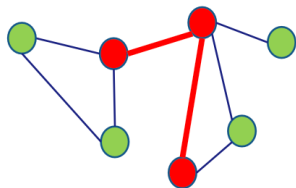


$$\exists \text{GNN}, s. t. ||\text{GNN} - f|| < \epsilon?$$

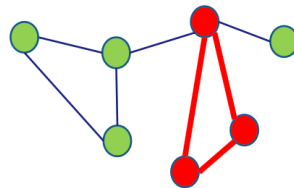
This kind of universality theorem has been proved for typical DNNs (Cybenko, 1989; Hornik, 1991)

# Một số bài toán mở

Are GNNs expressive enough to solve the following problems?



Finding the shortest path?



If a graph contains a circle?

**Yes**, if the **depth** and **width** are beyond certain bounds, with sufficiently discriminative node attributes (Loukas 2020)

# Tuần tới

- Nghỉ 30/04-01/05
- Tuần 10/05: Ví dụ về GNN trên đồ thị
- Tuần 17/05: Vấn đáp bài cuối kì