GCN

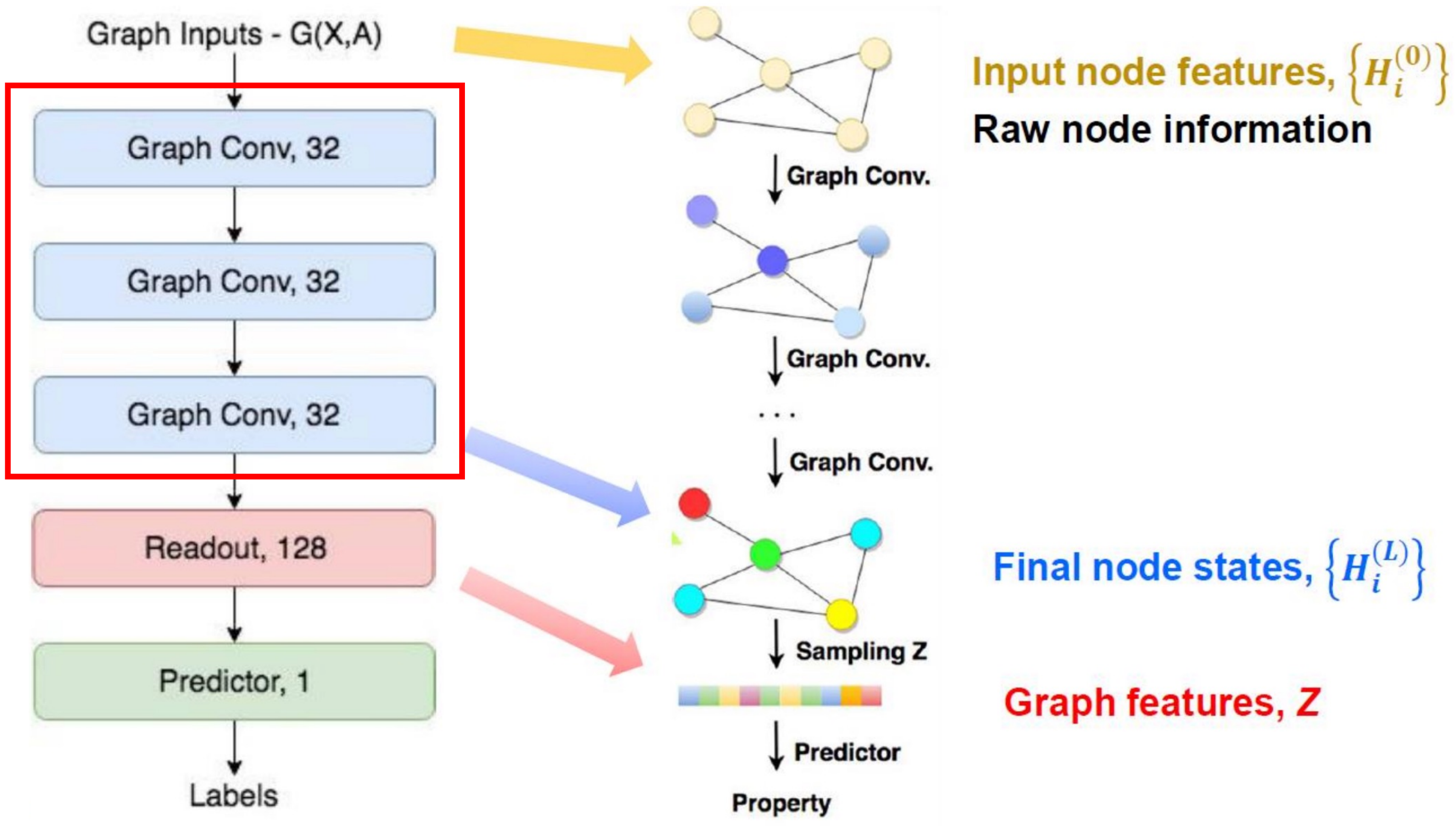
Graph Convolution Network

김태공 | 02.04.2023

Contents

1. Introduction

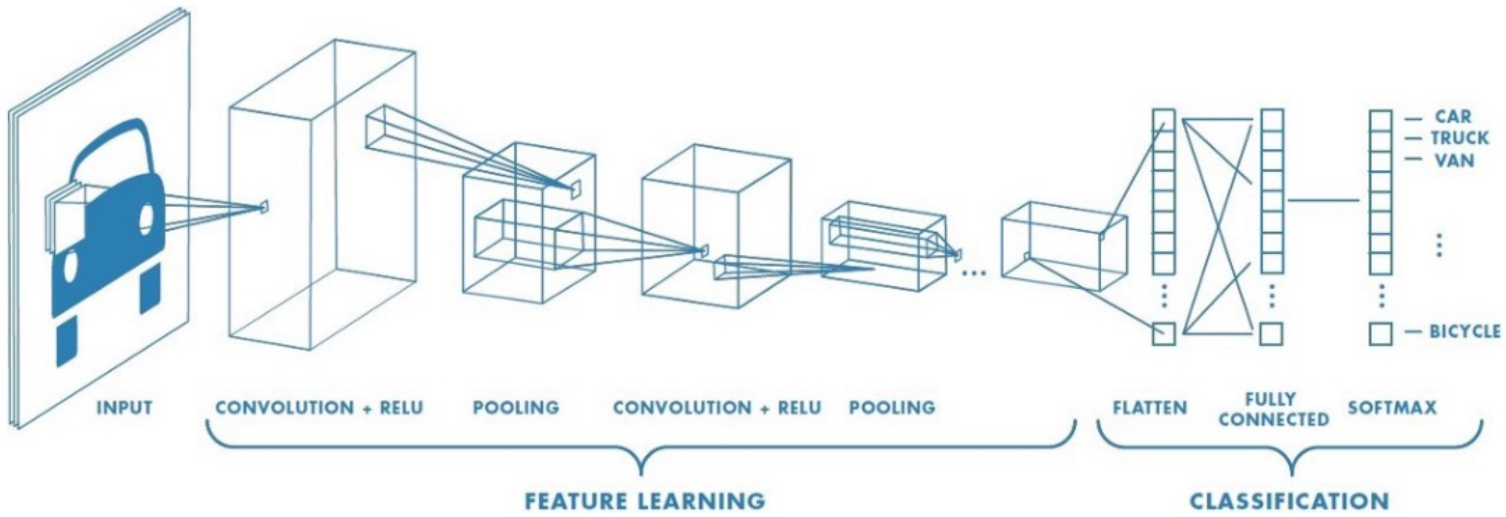
GCN Structure



2. Background

CNN Structure

Graph **Convolution Network**



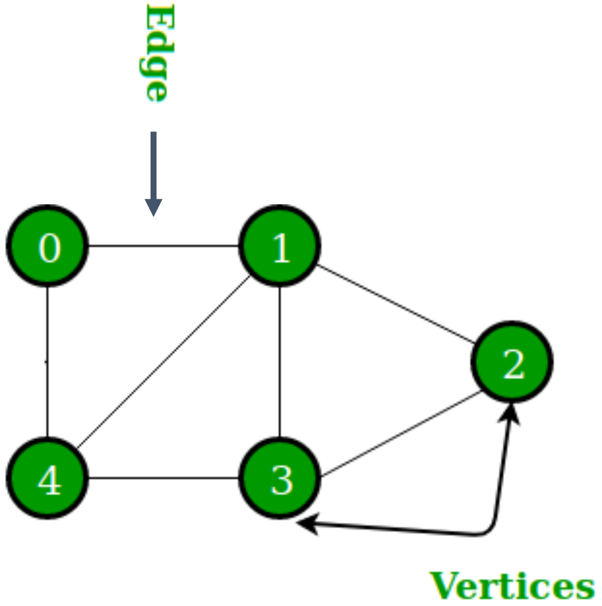
CNN Structure

- **Convolution, Pooling**
- Flatten, Fully Connected

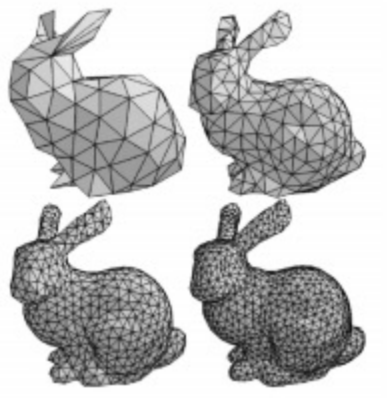
2. Background

Graph Structure

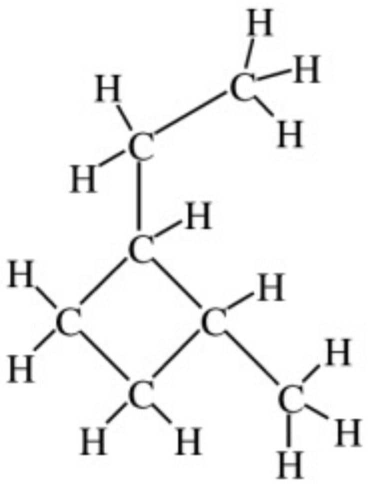
Graph Convolution Network



Social Graph



3D Mesh



Molecular Graph

Graph Components

- Edge: Relationship between two nodes
- Vertices(node): vertex of a graph

Graph Utilization

- Social structure, 3D image, **Molecular structure**
  - Vertices: atomic symbols (oxygen, hydrogen, …)
  - Edge: Molecular Bonds (ionic, covalent, …)

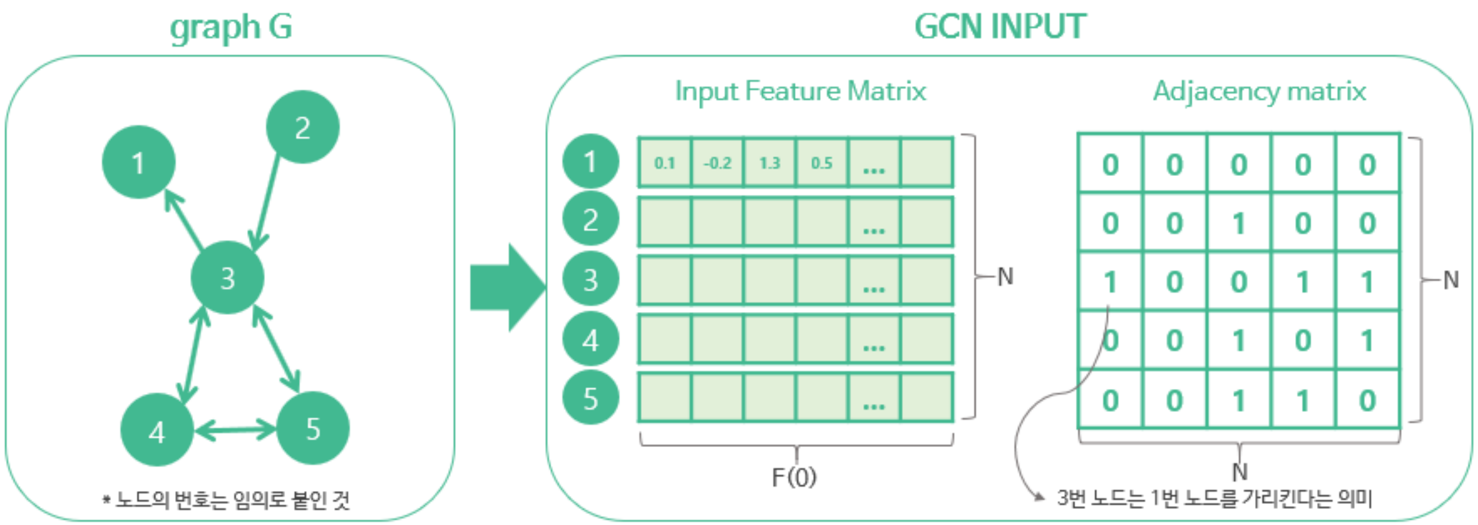
2. Background

Graph Structure

Graph Convolution Network

Graph Representation

- Adjacency Matrix
  - Connected information
  - $A_{ij} = relationship\ from\ node_i\ to\ node_j$
- Node Feature Matrix
  - Node information
  - $F_{ij} = j\text{-}th\ feature\ value\ in\ node_i$

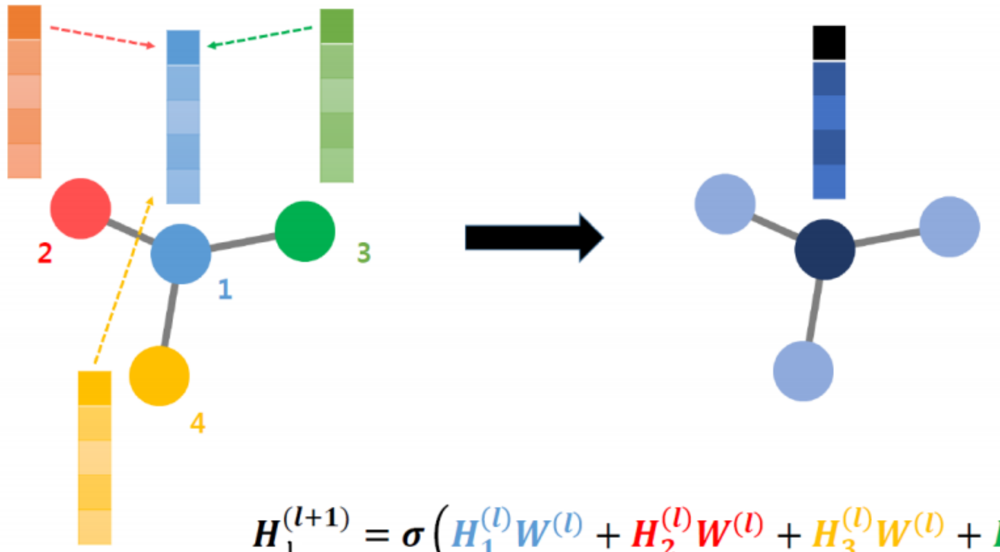


3. GCN

GCN Update (**GCN Conv**)

- Purpose
    - Node feature matrix update!!
    - Feature update affected by linked nodes
- Method
    - Feature update with connected parts in Graph
    - Similar to Conv layer operation
- Formula
    - $W: weight, l: l\text{-}th\ layer, H: Hidden\ state$
    - $H_1^{l+1}: node_1\ (l+1)\text{-}th\ node\ feature\ matrix$
    - H_1^{l+1}: affected by linked nodes ($node\ 1, 2, 3, 4$)

GCN Structure and Update



$$H_1^{(l+1)} = \sigma\left(H_1^{(l)}W^{(l)} + H_2^{(l)}W^{(l)} + H_3^{(l)}W^{(l)} + H_4^{(l)}W^{(l)} + b^{(l)}\right)$$

$$H_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} H_j^{(l)}W^{(l)} + b^{(l)}\right)$$

$where\ \sigma\ is\ a\ non-linear\ activation\ function$

3. GCN

GCN Update example (**GCN Conv**)

A

1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	1

H^1

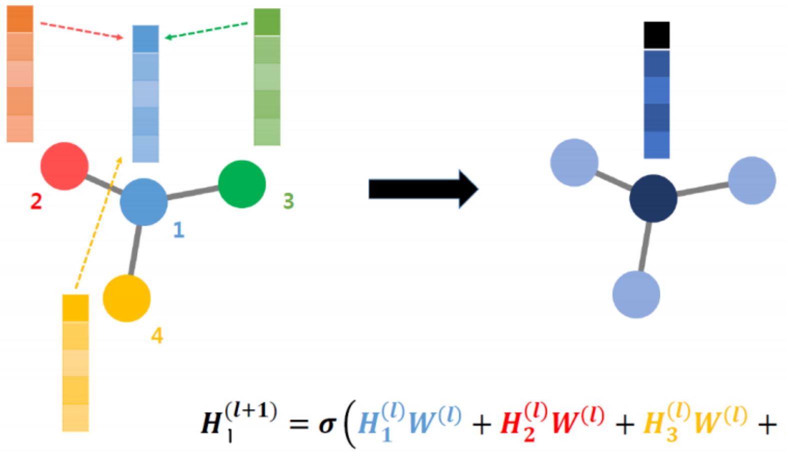
0.85	0.55	0.58	0.12	0.38
0.49	0.89	0.7	0.27	0.07
0.83	0.18	0.19	0.23	0.56
0.9	0.24	0.28	0.4	0.95

W^1

0.78	0.55	0.02	0.65	0.05	0.66
0.72	0.45	0.13	0.54	0.28	0.66
0.2	0.58	0.21	0.07	0.72	0.5
0.23	0.84	0.34	0.67	0.8	0.05
0.4	0.43	0.49	0.94	0.18	0.08

$$H^2 = AH^1W^1$$

5.1024	5.2401	1.9779	5.6482	3.1031	4.3366
2.6077	2.6485	0.8359	2.4225	1.7848	2.5303
2.4465	2.3973	0.8698	2.6582	1.292	2.0683
2.7574	2.8255	1.1468	3.2229	1.5833	2.2388



$$H_1^{(l+1)} = \sigma\left(H_1^{(l)}W^{(l)} + H_2^{(l)}W^{(l)} + H_3^{(l)}W^{(l)} + H_4^{(l)}W^{(l)} + b^{(l)}\right)$$

$$H_2^{(l+1)} = \sigma\left(H_1^{(l)}W^{(l)} + H_2^{(l)}W^{(l)} + b^{(l)}\right)$$

...

$$\Rightarrow H_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} H_j^{(l)}W^{(l)} + b^{(l)}\right)$$

$$H^{(l+1)} = \sigma\left(AH^{(l)}W^{(l)} + b^{(l)}\right)$$

learnable parameters are shared

3. GCN

GCN Update example (**GCN Conv**)

A

1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	1

H^2

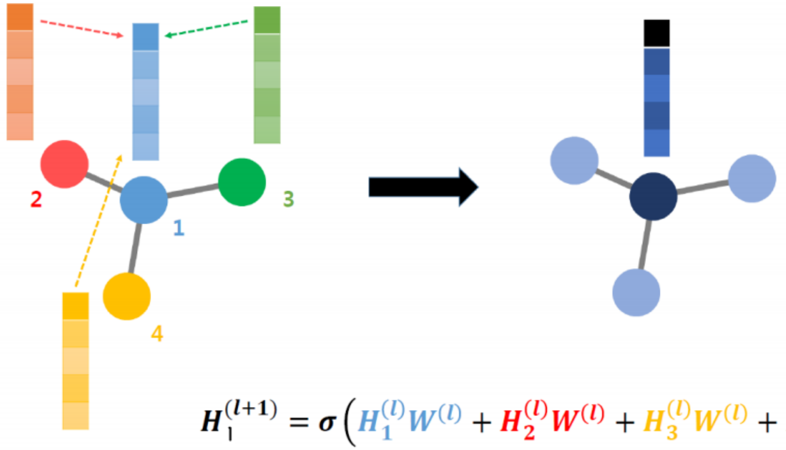
5.1024	5.2401	1.9779	5.6482	3.1031	4.3366
2.6077	2.6485	0.8359	2.4225	1.7848	2.5303
2.4465	2.3973	0.8698	2.6582	1.292	2.0683
2.7574	2.8255	1.1468	3.2229	1.5833	2.2388

W^2

0.71	0.38	0.57	0.93	0.48	0.86
0.56	0.36	0.25	0.23	0.45	0.02
0.14	0.05	0.19	0.37	0.93	0.1
0.68	0.01	0.92	0.17	0.65	0.72
0.66	0.24	0.9	0.58	0.13	0.11
0.91	0.53	0.19	0.23	0.57	0.75

$H^3 = AH^2W^2$

41.966	17.793	33.502	26.257	33.038	31.131
25.248	10.803	20.030	15.812	19.663	18.568
24.412	10.292	19.567	15.265	19.329	18.197
25.643	10.744	20.718	16.059	20.431	19.067



$$H_1^{(l+1)} = \sigma\left(H_1^{(l)}W^{(l)} + H_2^{(l)}W^{(l)} + H_3^{(l)}W^{(l)} + H_4^{(l)}W^{(l)} + b^{(l)}\right)$$

$$H_2^{(l+1)} = \sigma\left(H_1^{(l)}W^{(l)} + H_2^{(l)}W^{(l)} + b^{(l)}\right)$$

...

$$H_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} H_j^{(l)}W^{(l)} + b^{(l)}\right)$$

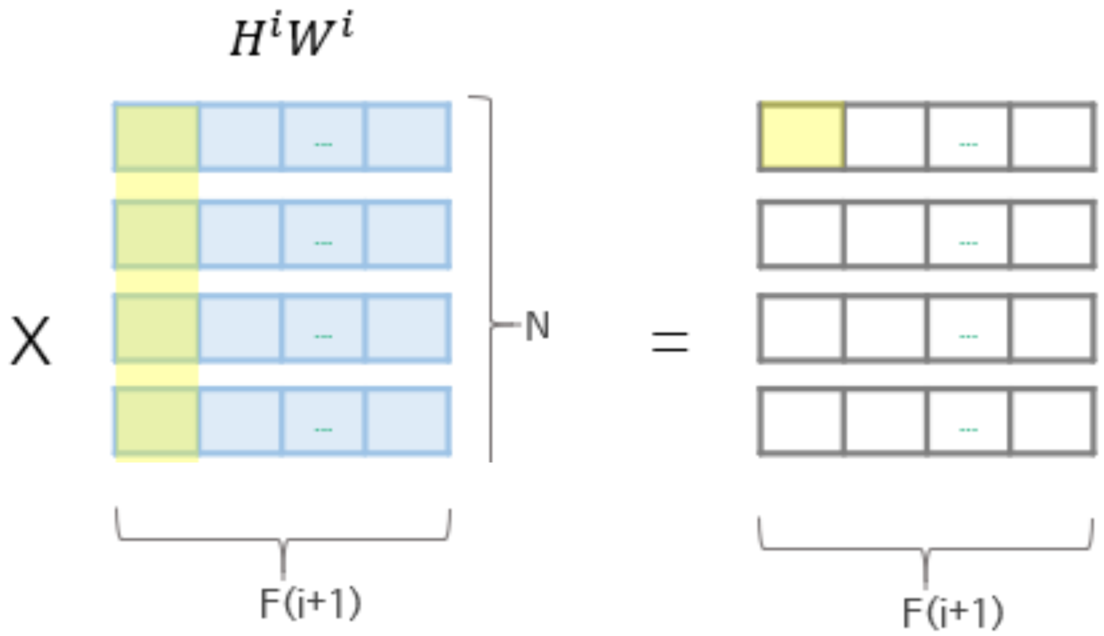
$$H^{(l+1)} = \sigma\left(AH^{(l)}W^{(l)} + b^{(l)}\right)$$

learnable parameters are shared

3. GCN

GCN Update (**GCN Conv**)

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



$$f(H^i, A) = \sigma(AH^iW^i)$$

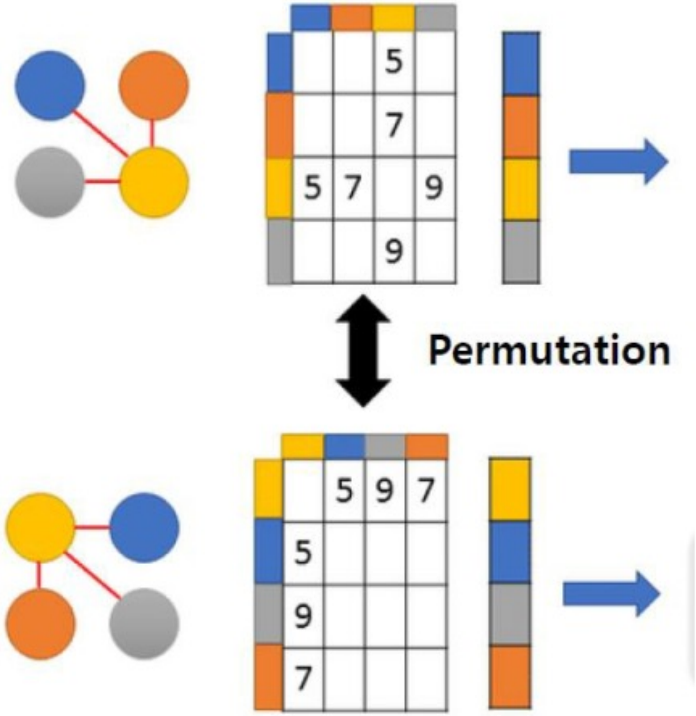
$where\ \sigma\ is\ a\ non-linear\ activation\ function$

3. GCN

Readout Layer

Readout Layer

- Purpose
  - Gather node information
  - Graphs of the same phase ➡️ Same graph
- Method
  - MLP operation on feature information for each node
  - Sum of all MLP operations



Node-wise summation

$$z_G = \tau\left(\sum_{i \in G} MLP\left(H_i^{(L)}\right)\right)$$

3. GCN

Readout Layer example

$H^3 = AH^2W^2$

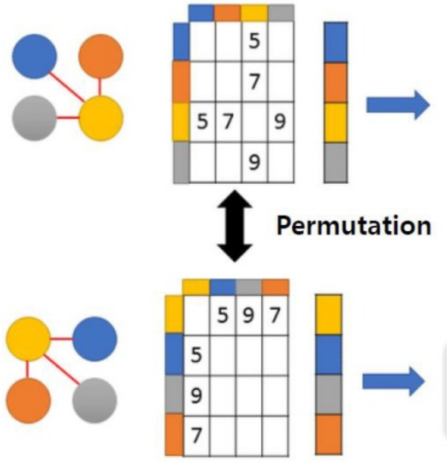
41.966	17.793	33.502	26.257	33.038	31.131
25.248	10.803	20.030	15.812	19.663	18.568
24.412	10.292	19.567	15.265	19.329	18.197
25.643	10.744	20.718	16.059	20.431	19.067

$W_{readout}$

0.71	0.38	0.57	0.93	0.48	0.86
0.56	0.36	0.25	0.23	0.45	0.02
0.14	0.05	0.19	0.37	0.93	0.1
0.68	0.01	0.92	0.17	0.65	0.72
0.66	0.24	0.9	0.58	0.13	0.11
0.91	0.53	0.19	0.23	0.57	0.75

$\mathrm{H}^3 W_{readout}$

$MLP(H_1^3)$	112.441	48.719	94.541	86.304	98.415	85.685
+						
$MLP(H_2^3)$	67.408	29.204	56.671	51.741	59.028	51.407
+						
$MLP(H_3^3)$	65.534	28.397	55.105	50.303	57.356	49.923
+						
$MLP(H_4^3)$	68.881	29.819	58.026	52.952	60.376	52.451
=						
$\sum MLP(H^3)$	314.265	136.139	264.342	241.300	275.174	239.467

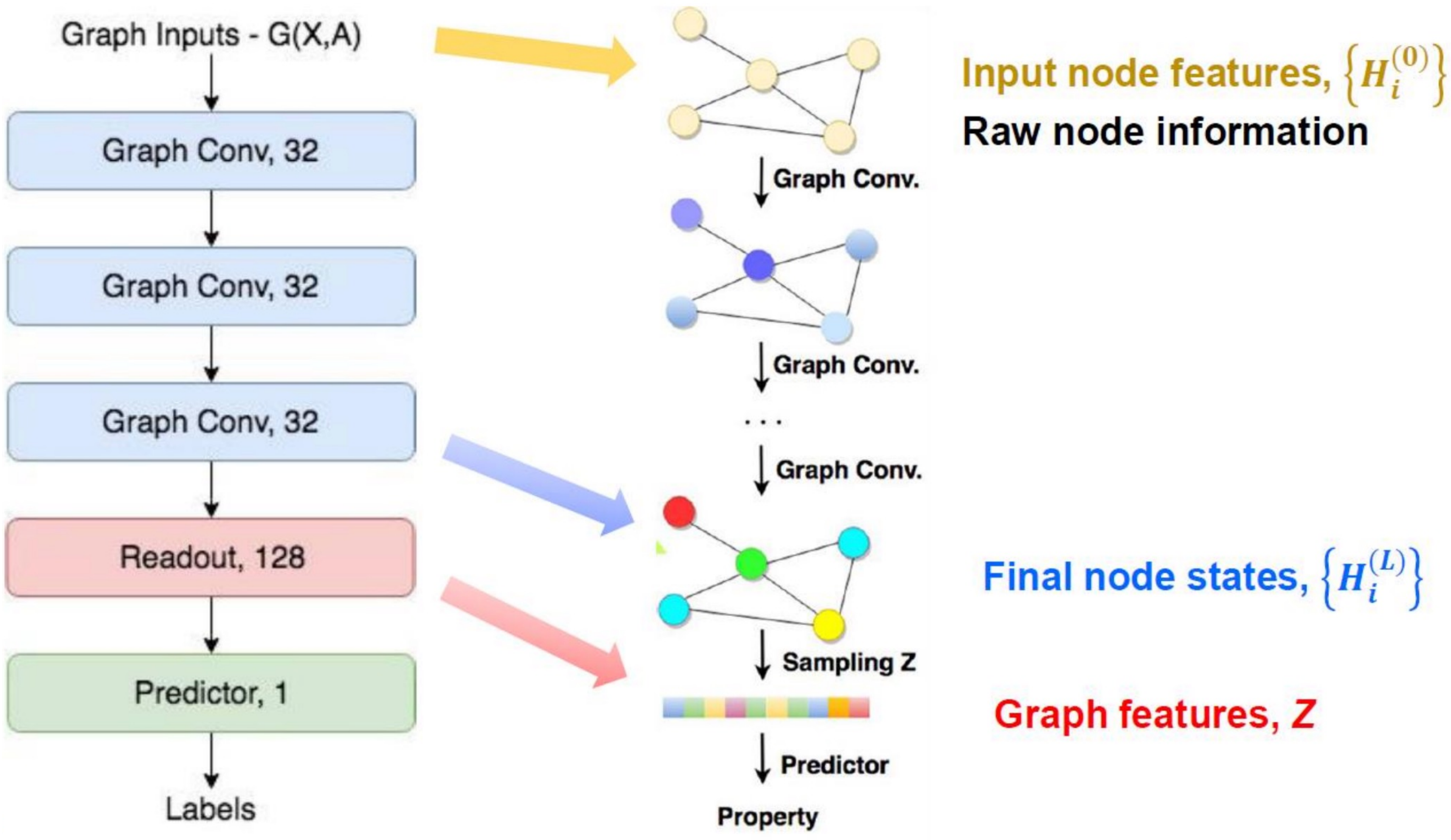


Node-wise summation

$$z_G = \tau\left(\sum_{i \in G} MLP\left(H_i^{(L)}\right)\right)$$

3. GCN

GCN Structure



4. GCN vs CNN

Common

- Conv Layer
  - Weight Sharing

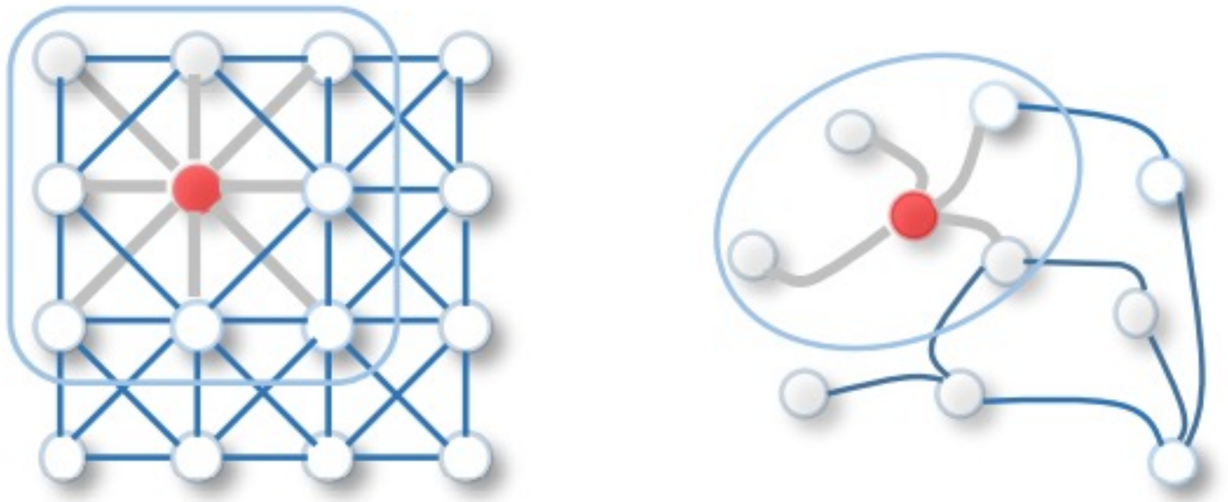
CNN

- Collect information on areas adjacent to specific pixels

GCN

- Collect information on nodes connected to the node

GCN vs CNN



4. GCN vs CNN

GCN Update example

A

1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	1

H^2

5.1024	5.2401	1.9779	5.6482	3.1031	4.3366
2.6077	2.6485	0.8359	2.4225	1.7848	2.5303
2.4465	2.3973	0.8698	2.6582	1.292	2.0683
2.7574	2.8255	1.1468	3.2229	1.5833	2.2388

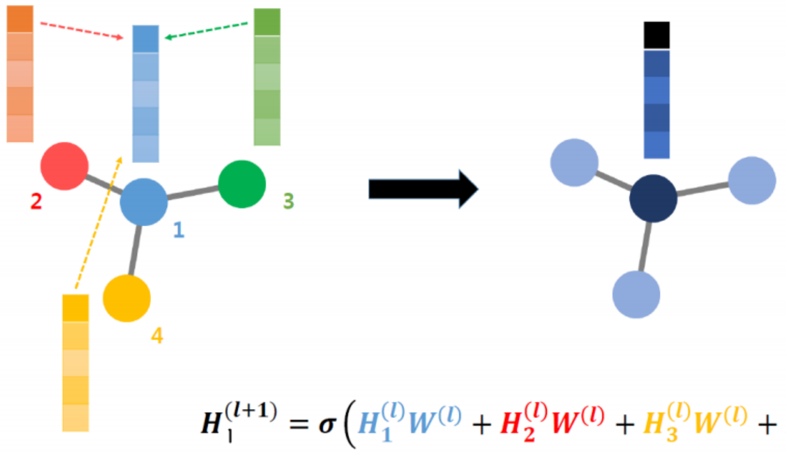
W^2

0.71	0.38	0.57	0.93	0.48	0.86
0.56	0.36	0.25	0.23	0.45	0.02
0.14	0.05	0.19	0.37	0.93	0.1
0.68	0.01	0.92	0.17	0.65	0.72
0.66	0.24	0.9	0.58	0.13	0.11
0.91	0.53	0.19	0.23	0.57	0.75

f by 1 filter

$H^3 = AH^2W^2$

41.966	17.793	33.502	26.257	33.038	31.131
25.248	10.803	20.030	15.812	19.663	18.568
24.412	10.292	19.567	15.265	19.329	18.197
25.643	10.744	20.718	16.059	20.431	19.067



$$H_1^{(l+1)} = \sigma\left(H_1^{(l)}W^{(l)} + H_2^{(l)}W^{(l)} + H_3^{(l)}W^{(l)} + H_4^{(l)}W^{(l)} + b^{(l)}\right)$$

$$H_2^{(l+1)} = \sigma\left(H_1^{(l)}W^{(l)} + H_2^{(l)}W^{(l)} + b^{(l)}\right)$$

...

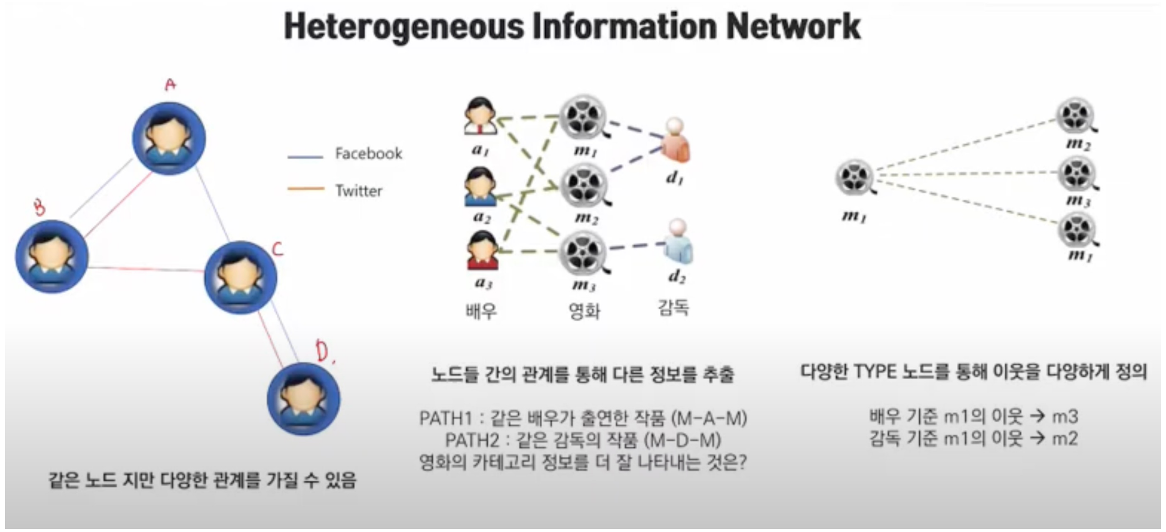
$$H_i^{(l+1)} = \sigma\left(\sum_{j \in N(i)} H_j^{(l)}W^{(l)} + b^{(l)}\right)$$

$$H^{(l+1)} = \sigma\left(AH^{(l)}W^{(l)} + b^{(l)}\right)$$

learnable parameters are shared

Next Presentation

Knowledge graph representation for recommendation



https://www.youtube.com/watch?v=prX2p9S8C1U

Reference

- Blog
    - https://signing.tistory.com/125
    - https://littlefoxdiary.tistory.com/17
    - https://ganghee-lee.tistory.com/27
- Github
    - https://github.com/heartcored98/Standalone-DeepLearning/blob/master/Lec9/Lab11_logP_Prediction_with_GCN.ipynb
    - https://github.com/SeungsuKim/CH485--AI-and-Chemistry/blob/master/Assignments/5.%20GCN/Assignment5_logP_GCN.ipynb
- Youtube
    - https://www.youtube.com/watch?v=YL1jGgcY78U
    - https://www.youtube.com/watch?v=9eMbvfRM9_8