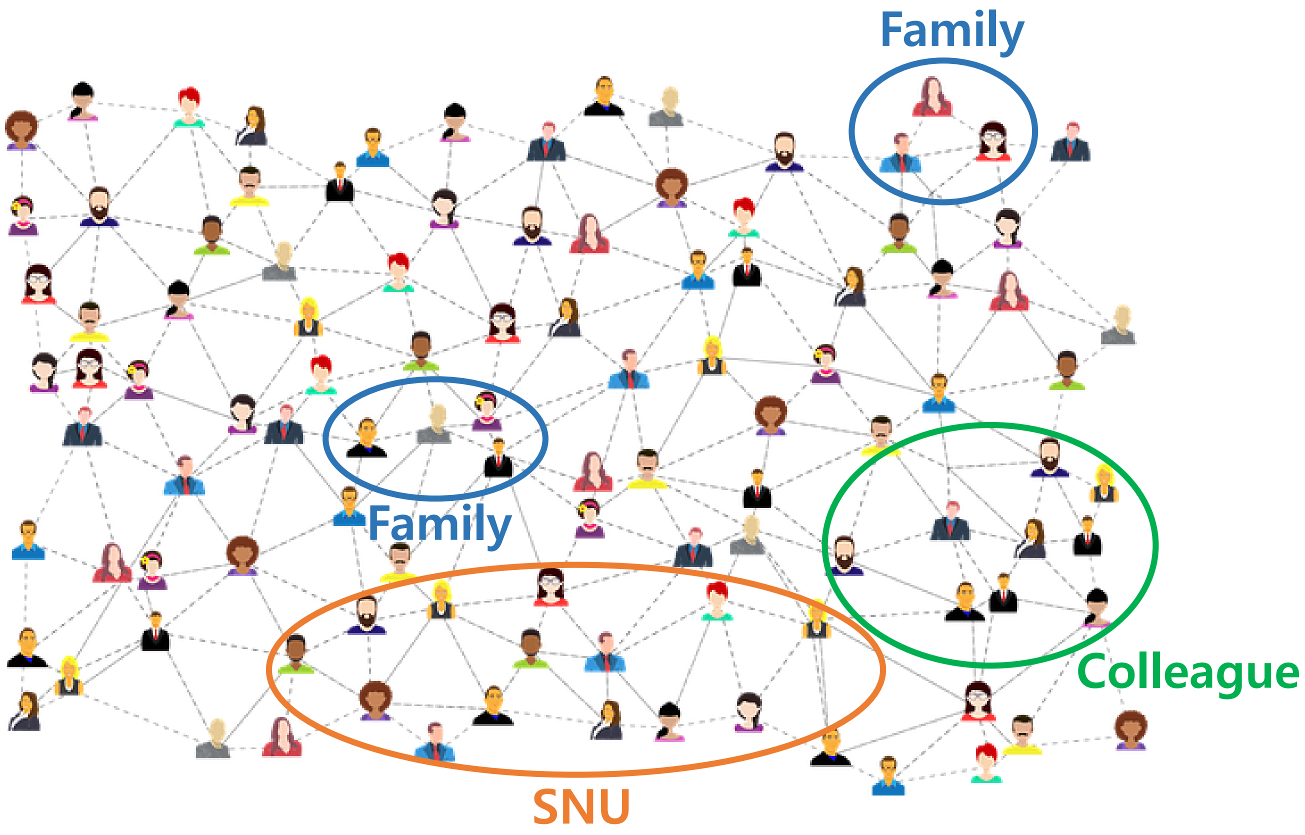
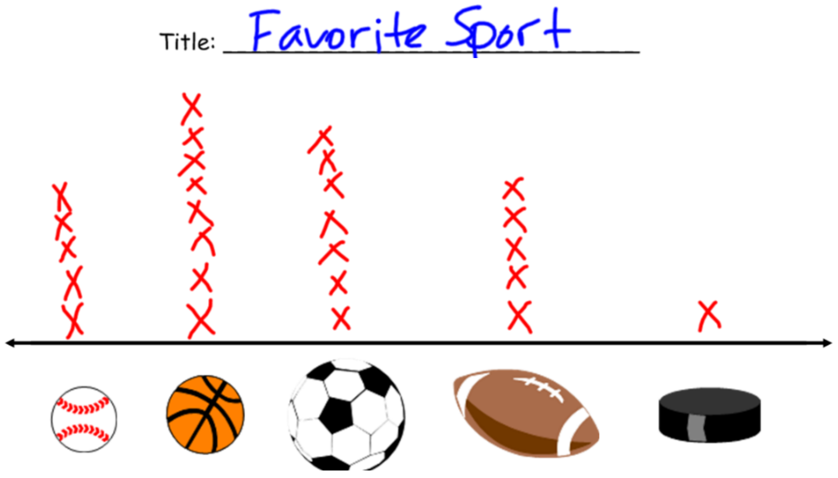
Hypergraph Neural Networks

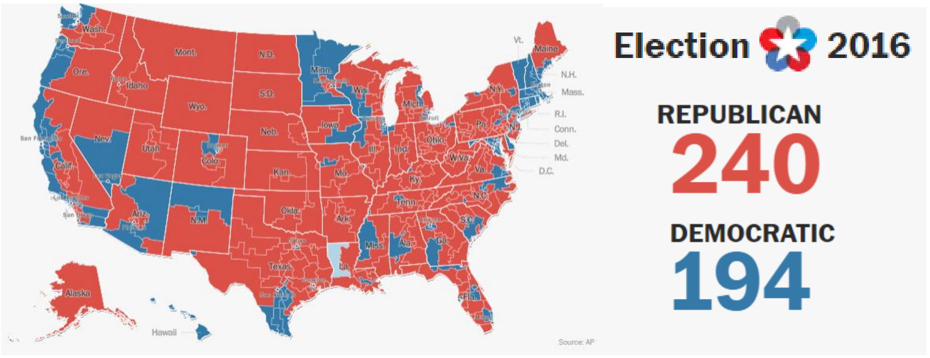
Yifan Feng et al. AAAI 2019

Presenter : Dae Ho Um

Data structure in real practice?

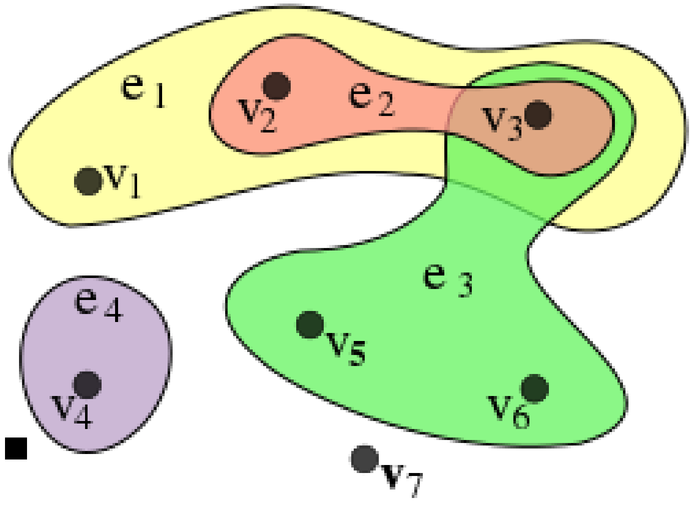






⇒ Often beyond pairwise connections!

Hypergraph



	e_1	e_2	e_3	e_4
v_1	1	0	0	0
v_2	1	1	0	0
v_3	1	1	1	0
v_4	0	0	0	1
v_5	0	0	1	0
v_6	0	0	1	0
v_7	0	0	0	0

- Hypergraph is a generalization of a graph in which an edge can connect any number of vertices.
- $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$
  - $\mathcal{V}$: a vertex set
  - $\mathcal{E}$: a hyperedge set
  - $\mathbf{W}$: diagonal matrix of edge weights
- $\mathbf{H}$: incidence matrix , $|\nu| \times |\varepsilon|$

$$h(v,e) = \begin{cases} 1, & \text{if } v \in e \\ 0, & \text{if } v \notin e, \end{cases}$$

HyperGraph Neural Networks

hypergraph Laplacian : Δ

$$\mathbf{\Theta} = \mathbf{D}_v^{-1/2}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{D}_v^{-1/2}$$
$$\mathbf{\Delta} = \mathbf{I} - \mathbf{\Theta}$$

spectral convolution using the truncated ChebyShev expansion

$$\mathbf{g} \star \mathbf{x} \approx \sum_{k=0}^{K} \theta_k T_k(\tilde{\mathbf{\Delta}})\mathbf{x} \quad (\tilde{\mathbf{\Delta}} = \frac{2}{\lambda_{max}}\mathbf{\Delta})$$

$$\lambda_{max} \approx 2 \text{ , } K = 1$$

$$\approx \theta_0\mathbf{x} - \theta_1\mathbf{D}^{-1/2}\mathbf{H}\mathbf{W}\mathbf{D}_\mathbf{e}^{-1}\mathbf{H}^\top\mathbf{D}_\mathbf{v}^{-1/2}\mathbf{x}$$

$$\begin{cases} \theta_1 = -\frac{1}{2}\theta \\ \theta_0 = \frac{1}{2}\theta\mathbf{D}_v^{-1/2}\mathbf{H}\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{D}_v^{-1/2} \end{cases}$$

$$\approx \frac{1}{2}\theta\mathbf{D}_v^{-1/2}\mathbf{H}(\mathbf{W}+\mathbf{I})\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{D}_v^{-1/2}\mathbf{x}$$
$$\approx \theta\mathbf{D}_v^{-1/2}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^\top\mathbf{D}_v^{-1/2}\mathbf{x},$$

hyperedge convolution

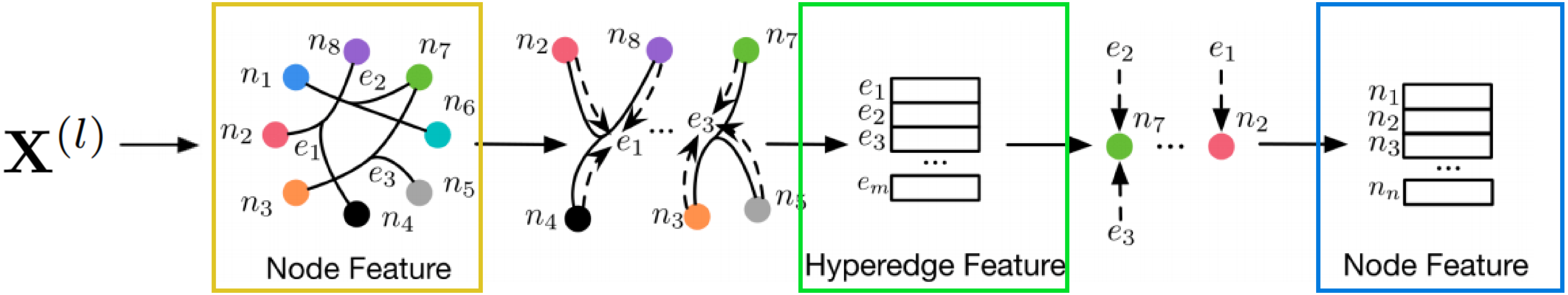
$$\Rightarrow \quad \mathbf{Y} = \mathbf{D}_\mathbf{v}^{-1/2}\mathbf{H}\mathbf{W}\mathbf{D}_\mathbf{e}^{-1}\mathbf{H}^\top\mathbf{D}_\mathbf{v}^{-1/2}\mathbf{X}\mathbf{\Theta}$$

HyperGraph Neural Networks

- hyperedge convolutional layer

$$\mathbf{X}^{(l+1)} = \sigma\left(\mathbf{D}_v^{-1/2}\mathbf{H}\mathbf{W}\mathbf{D}_e^{-1}\mathbf{H}^{\top}\mathbf{D}_v^{-1/2}\mathbf{X}^{(l)}\mathbf{\Theta}^{(l)}\right)$$

N by E $\quad$ E by N

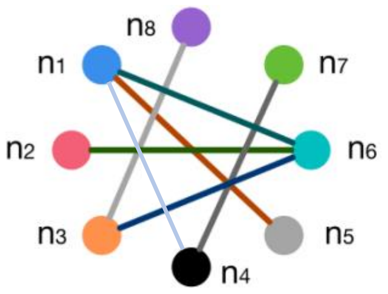


$\mathbb{R}^{N \times C_1}$ $\quad$ $\mathbb{R}^{N \times C_2}$ $\quad$ $\mathbb{R}^{E \times C_2}$ $\quad$ $\mathbb{R}^{N \times C_2}$

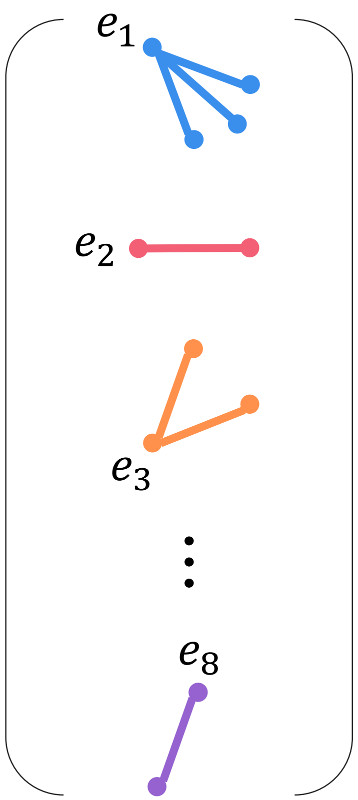
Experiments

- Citation network classification

- Hypergraph generation



pairwise graph, A



Hypergraph, H

- Results

Method	Cora	Pubmed
DeepWalk (Perozzi, Al-Rfou, and Skiena 2014)	67.2%	65.3%
ICA (Lu and Getoor 2003)	75.1%	73.9%
Planetoid (Yang, Cohen, and Salakhutdinov 2016)	75.7%	77.2%
Chebyshev (Defferrard, Bresson, and Vandergheynst 2016)	81.2%	74.4%
GCN (Kipf and Welling 2017)	81.5%	79.0%
HGNN	**81.6%**	**80.1%**

Table 2: Classification results on the Cora and Pubmed datasets.

Experiments

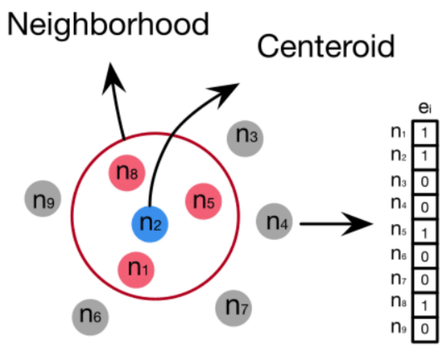
▪ Visual object classification

- A, Affinity Matrix (for GCN)

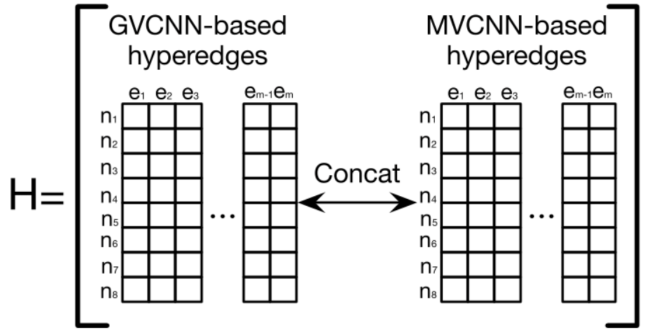
$$A_{ij} = \exp(-\frac{2Dij^2}{\Delta})$$

(D : Euclidean distance, Δ : Average Euclidean distance)

- H, Incidence Matrix (for HGNN)



Hyperedge Generation



Incidence Matrix Generation

- Results

Feature	Features for Structure					
	GVCNN		MVCNN		GVCNN+MVCNN	
	GCN	HGNN	GCN	HGNN	GCN	HGNN
GVCNN (Feng et al. 2018)	91.8%	**92.6%**	91.5%	**91.8%**	92.8%	**96.6%**
MVCNN (Su et al. 2015)	92.5%	**92.9%**	86.7%	**91.0%**	92.3%	**96.6%**
GVCNN+MVCNN	-	-	-	-	94.4%	**96.7%**

Table 4: Comparison between GCN and HGNN on the ModelNet40 dataset.

Method	Classification Accuracy
PointNet (Qi et al. 2017a)	89.2%
PointNet++ (Qi et al. 2017b)	90.7%
PointCNN (Li et al. 2018)	91.8%
SO-Net (Li, Chen, and Lee 2018)	93.4%
HGNN	**96.7%**

Table 6: Experimental comparison among recent classification methods on ModelNet40 dataset.

Conclusion

- HGNN is a more general framework which is able to handle the complex and high-order correlations through the hypergraph structure for representation learning compared with traditional graph.

- HGNN generalizes the convolution operation to the hypergraph learning process.

감사합니다