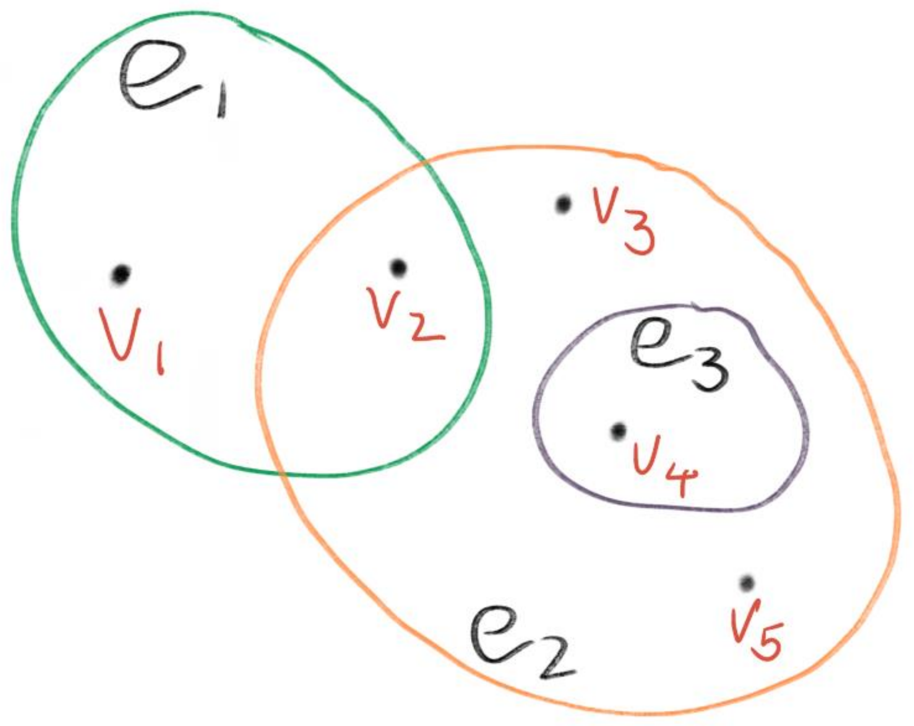
# HNHN
# Hypergraph Networks with Hyperedge Neurons

Yihe Dong (Microsoft)
Will Sawin (Columbia University)
Yoshua Bengio (Mila)

# Hypergraphs – definition

**Hypergraph**:
A graph generalization where one edge can join any number of nodes.




E.g: a hypergraph consisting of 3 hyperedges, $e_1$, $e_2$, and $e_3$ , and 5 hypernodes, $v_1$, $v_2$, $v_3$, $v_4$, and $v_5$.

# Hypergraphs – applications

- Represent co-authorship networks
  - Hyperedges: authors.
  - Hypernodes: papers.
- Represent co-citation data
  - Hyperedges: papers that are co-cited together.
  - Hypernodes: citing papers.

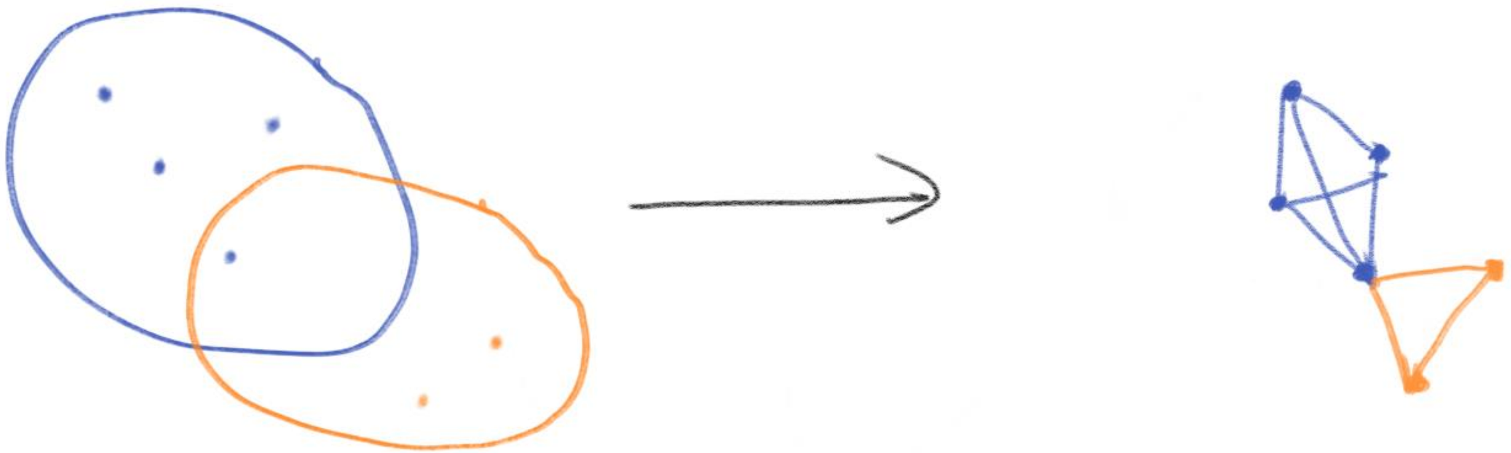
# Hypergraph representation learning

Prior work:

- GCN on hypergraph clique expansion [Zhou et al ‘07, Sun et al ‘08, Tu et al ‘18, Jian et al ‘18].
- GCN on hypergraph star expansion [Zien et al ’99, Sun et al ‘08].
- HyperGCN [Yadati et al ‘19].
- HGNN [Feng et al ‘18].
- HCHA [Bai et al ‘19].
- Many others.

# Limitations on existing approaches

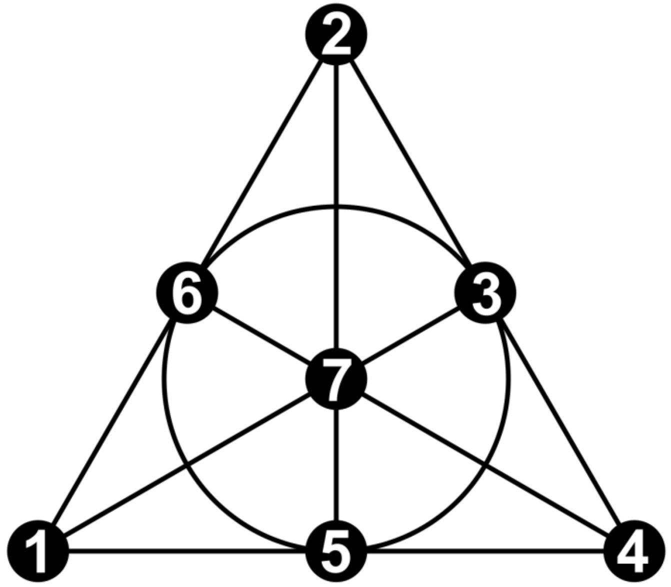
- Graph convolution on clique expansion of hypergraph.

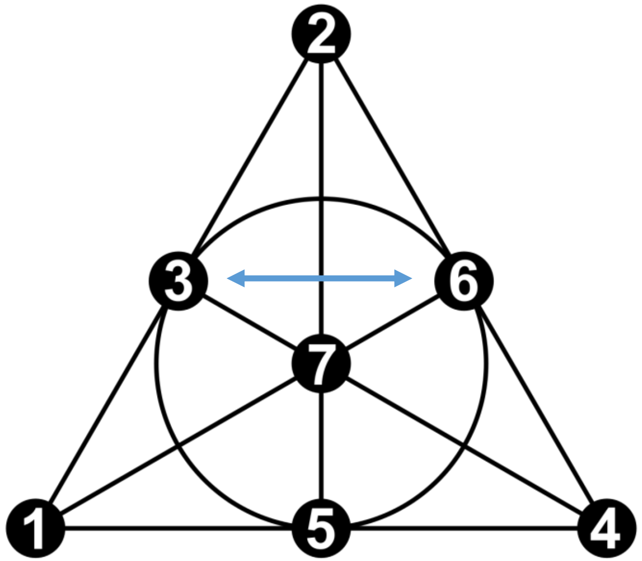


May lose hypergraph structural information.

# Limitations on existing approaches

- Hypergraph example: Fano plane

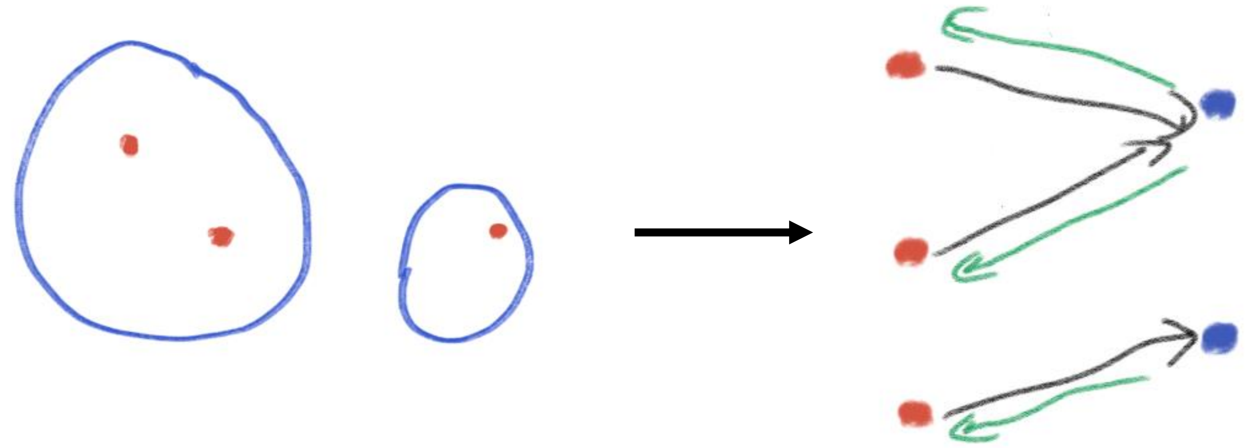



Produces **same** clique expansion despite hypernode permutation.

# Limitations on existing approaches

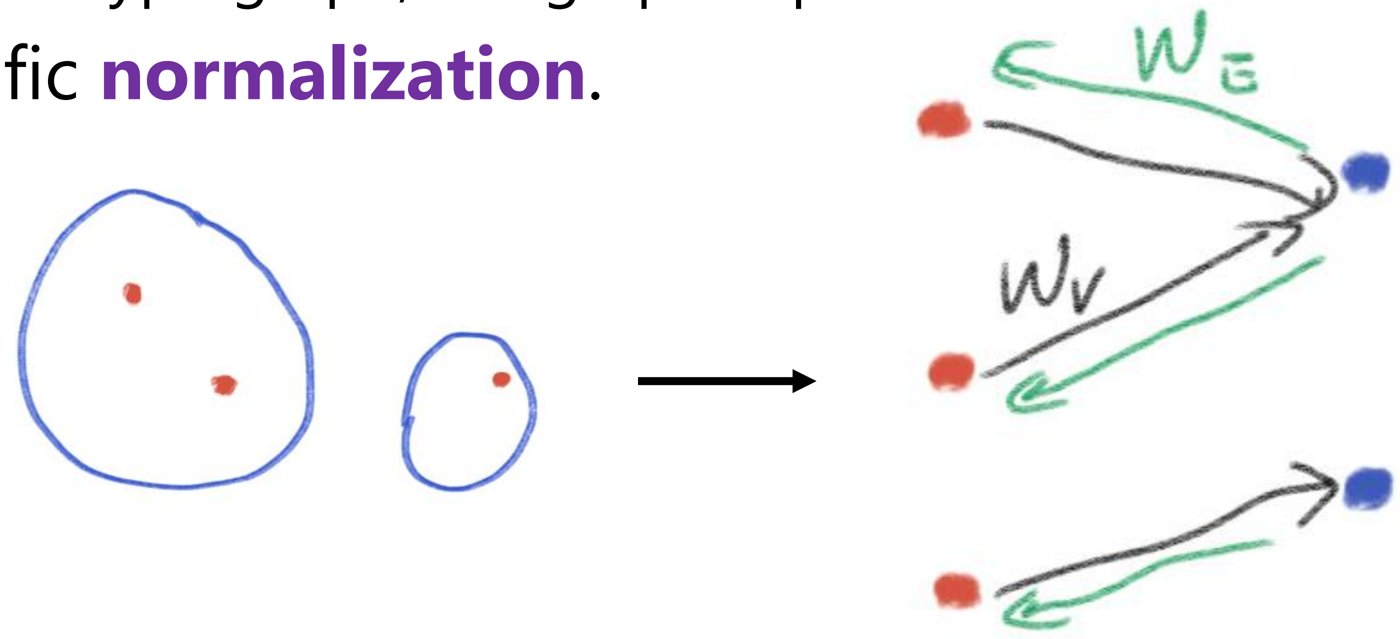
- Graph convolution on star expansion of hypergraph.



Treats hyperedges and hypernodes as the same.

# HNHN – architecture

- **Independent** weights and nonlinearities for hypernodes and hyperedges.
  - Convolution directly on hypergraph, not graph expansion.
- Flexible dataset-specific **normalization**.



# HNHN – architecture

- **Generalizes** both star and clique expansions:
  - Star expansion when $W_V = W_E$ .
  - Clique expansion when only keeping node nonlinearities.

# HNHN – normalization

- Hyperedge and hypernode normalization should **depend** on the **hyperedge degree** and **hypernode cardinality.**
  - Use normalization parameters $\alpha$ and $\beta$ to account for edge degree and node cardinality, respectively.

- Example: paper with fewer authors are more predictive of its authors' specialty.

# HNHN – normalization

- Compute node representation $X'_V$ from hyperedge representation $X_E$:

$$X'_V = \sigma\left(D_{V,l,\alpha}^{-1} A\, D_{E,r,\alpha}\, X_E\, W_V + b_V\right)$$

- $D_{V,l,\alpha}$ and $D_{E,r,\alpha}$ : normalization matrices depending on hyperedge cardinalities and normalization hyperparameter $\boldsymbol{\alpha}$.
- $W_V$, $b_V$: weights and bias.
- $A$ : vertex-edge incidence matrix.
- **Generalizes** normalization used in many prior works.

# Experimental datasets

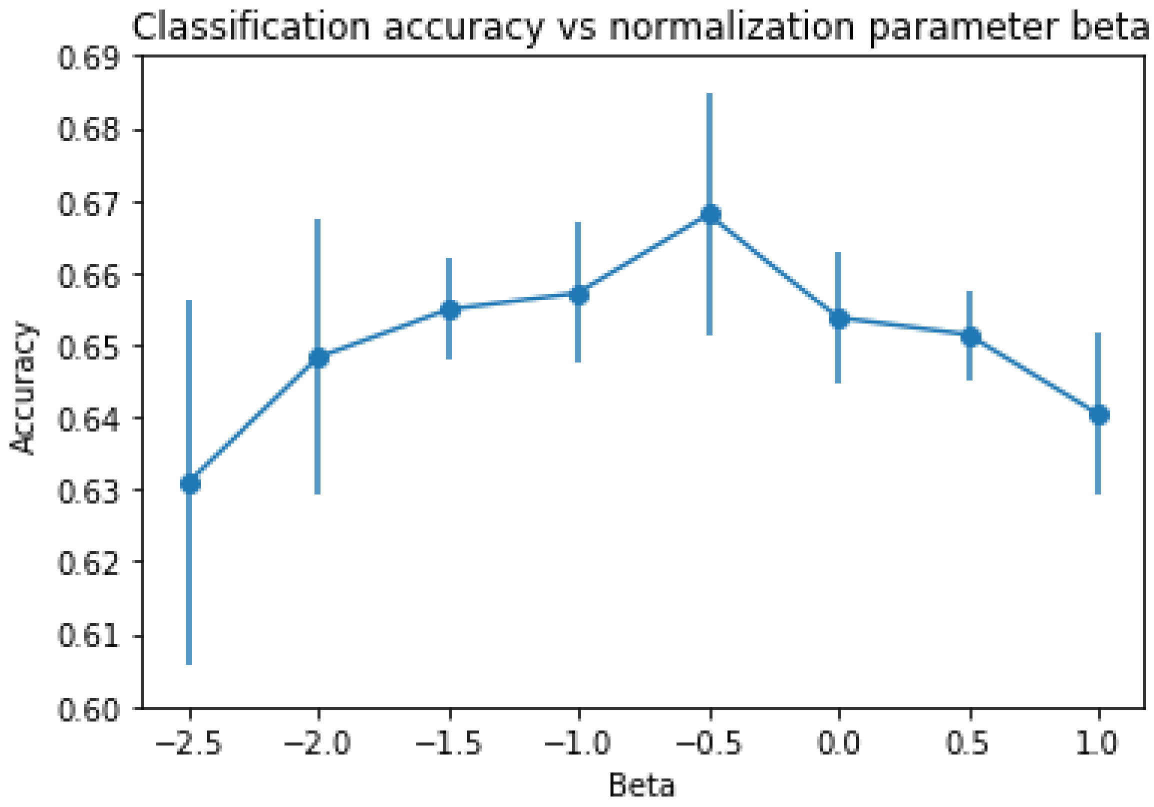
- co-authorship
  - Cora
  - DBLB
- Co-citation
  - CiteSeer
  - PubMed

# Hypernode classification

| | Accuracy | | | | Timing | | | |
|---|---|---|---|---|---|---|---|---|
| | DBLP | Cora | CiteSeer | PubMed | DBLP | Cora | CiteSeer | PubMed |
| HyperGCN | $71.3\pm1.2$ | $55.0\pm.9$ | $54.7\pm9.8$ | $60.0\pm10.7$ | $563.4\pm27.8$ | $183.4\pm2.7$ | $15.6\pm.2$ | $171.1\pm2.8$ |
| * Fast | $70.5\pm14.3$ | $45.2\pm12.9$ | $56.1\pm11.2$ | $54.4\pm10.0$ | $\mathbf{11.5\pm.1}$ | $\mathbf{2.9\pm.1}$ | $\mathbf{1.1\pm0.}$ | $\mathbf{2.5\pm.1}$ |
| HGNN | $77.6\pm.4$ | $58.2\pm.3$ | $61.1\pm2.2$ | $63.3\pm2.2$ | $802.9\pm59.2$ | $298.4\pm12.2$ | $30.5\pm.8$ | $270.1\pm10.5$ |
| HNHN | $\mathbf{85.1\pm.2}$ | $\mathbf{63.9\pm.8}$ | $\mathbf{64.8\pm1.6}$ | $\mathbf{75.9\pm1.5}$ | $44.2\pm1.3$ | $13.6\pm5.4$ | $1.3\pm.1$ | $26.6\pm.4$ |

- Node classification accuracy and timing results on various datasets compared to SOTA methods.
- * Fast stands for HyperGCN Fast.
- Accuracy measured in %, timing measured in seconds.

# Dependence on normalization scheme




- $\beta = 0$ not necessarily optimal.
- Best $\beta$ < 0 → paper with fewer authors more predictive of its authors' field.

# Thank you

- Code: github.com/twistedcubic/HNHN